

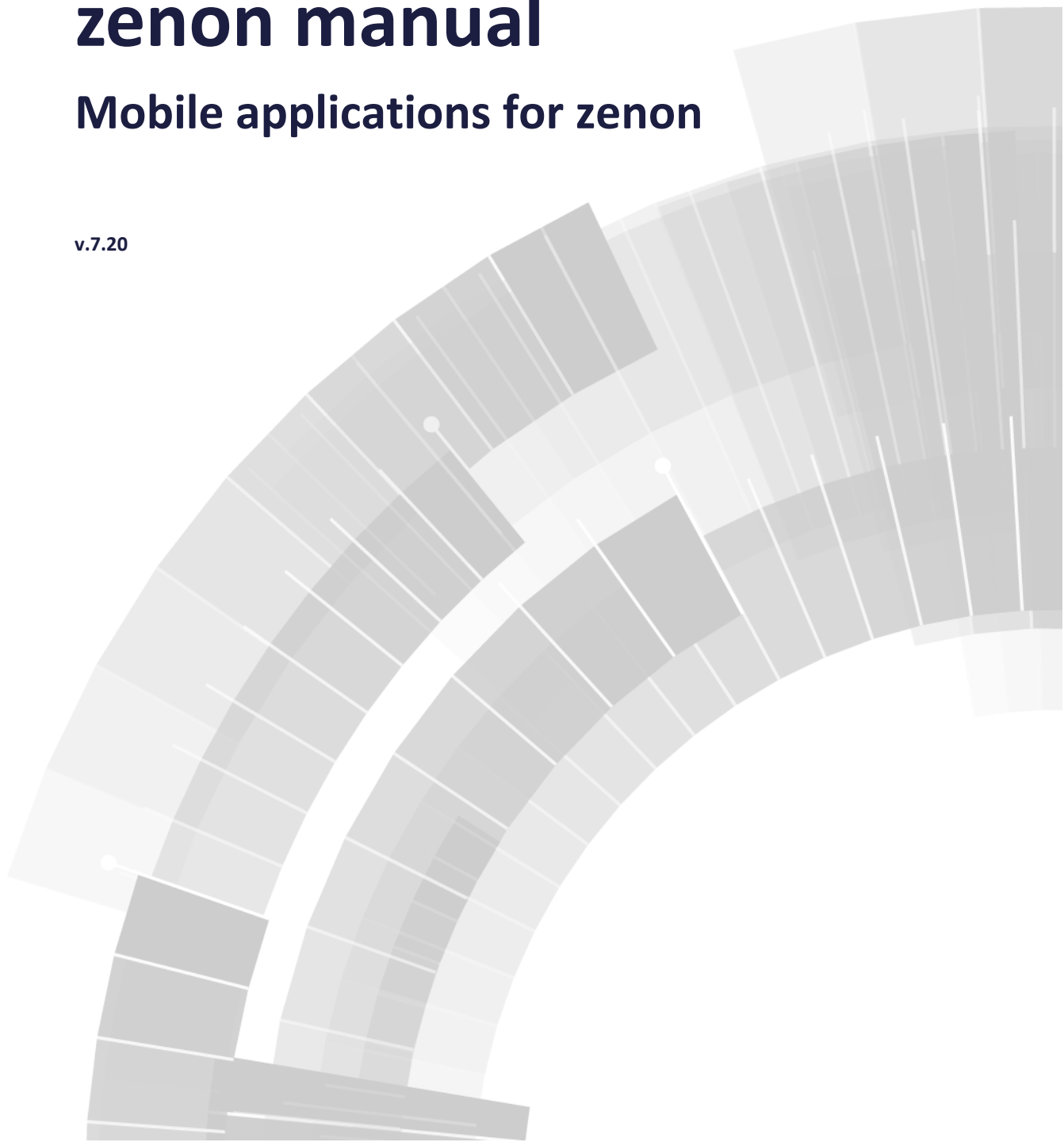


**COPADATA**  
do it your way

# zenon manual

## Mobile applications for zenon

v.7.20





©2015 Ing. Punzenberger COPA-DATA GmbH

All rights reserved.

Distribution and/or reproduction of this document or parts thereof in any form are permitted solely with the written permission of the company COPA-DATA. Technical data is only used for product description and are not guaranteed qualities in the legal sense. Subject to change, technical or otherwise.

# Contents

<b>1. Welcome to COPA-DATA help .....</b>	<b>6</b>
<b>2. Mobile applications for zenon .....</b>	<b>6</b>
<b>3. Everywhere Server by zenon .....</b>	<b>7</b>
3.1 Licensing .....	8
<b>4. Everywhere Server by zenon .....</b>	<b>8</b>
4.1 Requirements .....	8
4.2 Data security .....	9
4.3 Configuration .....	10
<b>5. zenon6.ini entries: .....</b>	<b>10</b>
5.1.1 Configuration with the Everywhere.Config.exe .....	11
5.1.2 User authorization .....	13
5.1.3 Everywhere - Certificate creator .....	13
5.1.4 Create root certificate .....	15
5.1.5 Certificate on the client .....	16
5.1.6 Certificates .....	17
5.2 CEL entries .....	19
5.3 Data model of the Everywhere Server .....	20
5.3.1 Data model - static project data .....	21
5.3.2 Data model - project data that changes .....	23
5.3.3 Querying archives .....	25
5.3.4 Calling up the data model using a web browser .....	26
5.3.5 Examples .....	28
5.3.6 Using the app on a mobile device (Android) .....	33
<b>6. Everywhere App by zenon .....</b>	<b>35</b>
<b>7. The following Apps are available: .....</b>	<b>36</b>
7.1 Requirements .....	36
7.2 Installation .....	36
7.2.1 Note for OEM user: .....	37

7.3	Everywhere App by zenon for MS Windows Phone.....	38
7.3.1	Authentication.....	38
7.3.2	Profile List .....	41
7.3.3	Edit Profile .....	44
7.3.4	Equipment Model.....	47
7.3.5	Equipment Groups .....	50
7.3.6	Variables selection .....	53
7.3.7	Values view.....	56
7.3.8	Alarm List.....	59
7.3.9	Acknowledge alarms .....	62
7.4	Everywhere App by zenon for iPhone and iPad .....	65
7.4.1	Authentication.....	65
7.4.2	Profile List .....	68
7.4.3	Edit Profile .....	70
7.4.4	Equipment Model.....	72
7.4.5	Equipment groups and variables.....	74
7.4.6	Variables selection .....	76
7.4.7	Values view.....	78
7.4.8	Alarm List.....	80
7.4.9	Acknowledge alarms .....	82
7.5	Everywhere App for Desktop .....	85
7.5.1	Authentication.....	85
7.5.2	Profile List .....	86
7.5.3	Edit Profile .....	88
7.5.4	Equipment model and equipment groups .....	90
7.5.5	Variables selection .....	91
7.5.6	Values view.....	92
7.5.7	Alarm List.....	94
7.5.8	Acknowledge alarms .....	96
7.6	Everywhere App by zenon for Android .....	97
7.6.1	Authentication.....	97
7.6.2	Profile List .....	99
7.6.3	Edit Profile .....	100
7.6.4	Equipment model and equipment groups .....	102
7.6.5	Variables selection .....	103
7.6.6	Values view.....	105

7.6.7	Acknowledging alarm list and alarms.....	106
<b>8.</b>	<b>Notifier App by zenon.....</b>	<b>108</b>
8.1	Functionality .....	108
8.2	Configuration .....	109
8.2.1	Configure keywords.....	115
8.2.2	Configure telephone numbers .....	118
8.2.3	Configure PIN and NA code .....	122
8.3	Receive and acknowledge alarm SMS.....	125

# 1. Welcome to COPA-DATA help

## GENERAL HELP

If you cannot find any information you require in this help chapter or can think of anything that you would like added, please send an email to [documentation@copadata.com](mailto:documentation@copadata.com) (<mailto:documentation@copadata.com>).

## PROJECT SUPPORT

You can receive support for any real project you may have from our Support Team, who you can contact via email at [support@copadata.com](mailto:support@copadata.com) (<mailto:support@copadata.com>).

## LICENSES AND MODULES

If you find that you need other modules or licenses, our staff will be happy to help you. Email [sales@copadata.com](mailto:sales@copadata.com) (<mailto:sales@copadata.com>).

# 2. Mobile applications for zenon

With the mobile applications, you can visualize your zenon projects on mobile end devices, tablets and desktop computers. The basis of this display is an existing zenon project. There is no additional project configuration work required.

For display on mobile end devices, tablets and desktop computers, there are zenon apps available in the respective app store.

You can also create your own mobile applications for the visualization of your zenon project configuration. You can find further information and examples of code in the Everywhere Server data model (on page 20) chapter.

#### THE FOLLOWING APPS ARE AVAILABLE:

- ▶ **Notifier-App by zenon** (on page 108):  
Allows the receipt and acknowledgment of SMS messages that are sent by the Message Control module.  
This App is available for the Android operating system.
- ▶ **Everywhere Server by zenon** (on page 7):  
Provides zenon data for mobile applications.
- ▶ **Everywhere App by zenon**:  
This app allows simple access to zenon via smartphones, tablets and desktop computers.  
The following apps are available:
  - **Everywhere App by zenon** (on page 38) for Windows Mobile
  - **Everywhere App by zenon** (on page 65) for iPhone, iPad product range
  - **Everywhere App by zenon** (on page 85) for Windows Desktop
  - **Everywhere App by zenon** (on page 97) for Android

Once a server has been configured, the following functions are available via the app:

- ▶ Real-time display of values of a zenon project
- ▶ Authentication by means of the mobile application
- ▶ Selection of the equipment model of the active project
- ▶ Assignment of the configured user authorizations (via .INI entry)
- ▶ Individual variables can be activated
- ▶ Display of values in lists ...
  - ... with graphic progress bars
  - ... with dynamic pointer instruments
  - ... Alarm messages with time

## 3. Everywhere Server by zenon

The Everywhere Server by zenon is for visualization of real-time data of a zenon project configuration on smartphones, tablets and desktop computers.

It runs in the context of zenon Runtime and is configured in zenon6.ini. No additional program need be started for configuration.



### License information

*The Everywhere Server by zenon must be licensed.*

*Any device that supports the OData protocol can be used as a client.*

*In addition to its programming, an app is provided for smartphones, tablets and desktop computers.*

*These apps are available for free in the respective app stores.*

## 3.1 Licensing

# 4. Everywhere Server by zenon

**Everywhere Server** gets the license of the Runtime. The license for Everywhere Server is issued by Runtime. The **Everywhere Server** can not be used, If the Runtime does not have a corresponding license.

For this reason, it may happen that the Runtime is running, but the **Everywhere Server** can not be started because of missing license rights. In this case, please use the licensing tool in order to receive a corresponding license.

## 4.1 Requirements

To use the server for use with mobile applications, the following requirements must be met:

- ▶ zenon from version 7.11 on
- ▶ Users have to be configured in the project.  
For details about user administration, see also the Elements in the report area chapter.
- ▶ For communication via HTTPS, the corresponding port must be enabled on the Server and the Client. This is port 8050 by default.



- ▶ zenon Runtime must also be running on the computer on which Everywhere Server by zenon is running.  
This computer/server must be contactable in the Internet or internal network.

#### REQUIREMENT FOR THE USE OF EVERYWHERE APP BY ZENON

The following requirements need only be met if the zenon project is visualized with Everywhere App by zenon:

- ▶ An equipment model must be configured in the project.  
Variables can be linked from zenon 7.11 at each level.  
You can find details on the configuration of an equipment model in the Equipment modeling chapter.
- ▶ zenon Runtime must also be running on the computer on which Everywhere Server by zenon is running.  
The computer/server must be contactable in the Internet. To do this, the computer must have a public IP address.  
**Attention:** 192.168.xxx.xxx is not a public address!

## 4.2 Data security

Secure transfer of data via HTTPS is ensured by the use of the Transport Layer Security encryption protocol (TLS). The identity of the Server is checked by the Client. This guarantees the integrity, origin and target of the data transfer.

#### OVERVIEW OF COMMUNICATION

- ▶ Everywhere Server by zenon uses a server certificate.  
Everywhere Server is not started without a server certificate. This server certificate is created with the `Everywhere - Certificate creator.exe` tool.  
If several servers are in use, separate server certificates can be used for each server.
- ▶ The server certificate is based on a root certificate.  
If several servers are in use, separate server certificates can be used for each server.  
These server certificates originate from the same root certificate.  
**Advantage:** different servers can be called up with a single root certificate.
- ▶ The root certificate can be purchased or you can create one yourself.  
Root certificates are also called SLL certificates.  
**Tip:** If a root certificate or SLL certificate is purchased from a provider, there is no installation on the Client. These certificates are installed together with the installation of the operating system.

## 4.3 Configuration

Everywhere Server by zenon is integrated into zenon Runtime. No additional program needs to be started. The necessary settings are made in zenon6.ini directly.



### Hint

*All .ini entries can also be made with the `Everywhere.Config.exe` (on page 11) program. This program offers a graphical user interface. The configurations are also transferred to zenon6.ini.*

## 5. zenon6.ini entries:

- ▶ Open zenon6.ini in an text editor.  
**Note:** You can find zenon6.ini under the following path:  
`%ProgramData%\COPA-DATA\System`
- ▶ Insert the server entries:

Entry	Description
[EVERYWHERE]	Configuration of the global properties for Everywhere Server.
CERTIFICATE=	<p>Server certificate subject for HTTPS</p> <p>Subject of the certificate. The certificate must be saved in the <b>Machine Store</b> in the <b>MY</b> node (your own certificates).</p> <p>e.g.: <b>CERTIFICATE=CN=MyComputerName</b></p>
ENABLE=	<ul style="list-style-type: none"> <li>▶ 1: Everywhere Server is started with zenon Runtime.</li> <li>▶ 0: Everywhere Server is not started.</li> </ul> <p>Default: 0</p>
PORT=	<p>HTTPS port for communication with the Everywhere Server</p> <p>Default: 8050</p>
WRITE_ACCESS=	<ul style="list-style-type: none"> <li>▶ 0: Only read access via Everywhere Server is possible.</li> <li>▶ 1: Allows the writing of variables and the acknowledgment of alarms.</li> </ul> <p>Default: 0</p>

Due to the system, only ANSI and Unicode are supported for reading the INI files.



### Attention

UTF-8 format is not supported!

You should therefore always save your INI files as a text file in ANSI or Unicode format.



### Hint

You can find further information on .ini entries in the Configuration files manual.

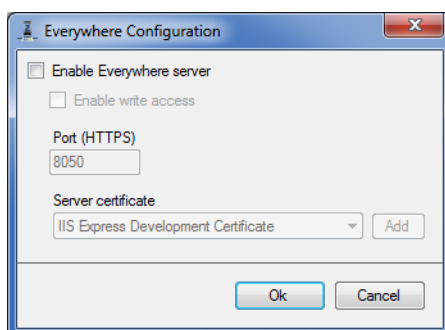
## 5.1.1 Configuration with the Everywhere.Config.exe

The .ini entries for the Everywhere Server can also be configured using the **Everywhere.Config.exe** program. You can find this program in the folder `C:\Program Files (x86)\Common Files\COPA-DATA\STARTUP`.

### Information

*The program **Everywhere.Config.exe** is only available in English.*

Double-clicking on the program opens the configuration dialog:



Parameters	Description
<b>Enable Everywhere server</b>	<p>If the checkbox is active, the Everywhere Server by zenon starts when zenon Runtime is started.</p> <p>Default: <code>inactive</code></p> <p>Note: If this checkbox is not active, all other settings are also inactive and grayed out.</p>
<b>Enable Everywhere write access</b>	<p>Activates write access to variables and alarms (acknowledgment) by the server.</p> <p>0 : read access only</p> <p>1 : Writing to variables and acknowledgment of alarms possible</p> <p>Default: 0</p>
<b>HTTPS port:</b>	<p>HTTPS port that is used by the Everywhere Server by zenon.</p> <p>Default: 8050</p>
<b>Server certificate</b>	<p>Server certificate for HTTPS communication. This can be selected in the drop-down list.</p> <p>This drop-down list contains all available certificates.</p> <p>Note: The certificate must contain a private key. If no server certificate is selected, Everywhere Server is not started! Certificates can be created with the <b>Everywhere.CertificateCreator.exe</b> tool.</p>
<b>Add</b>	<p>The <b>Everywhere.CertificateCreator.exe</b> (on page 13) program is opened by clicking on the button. This program is used for the creation of separate server certificates.</p> <p>The entry in the drop-down list is based on the <b>Friendly name</b> of the <b>Everywhere.CertificateCreator.exe</b></p>
<b>OK</b>	Applies settings and closes the dialog.
<b>Cancel</b>	Discards all changes and closes the dialog.

*Note: These configurations are saved in zenon6.ini.*

*You can find further information about these .ini entries in the Configuration files manual in the Everywhere Server [EVERYWHERE] chapter*



### Attention

*The configuration is also applicable for all zenon installations on one computer.*

### 5.1.2 User authorization

The assignment of configured user authorizations is possible via the zenon6.ini file or in the zenon Editor.

#### IN THE EDITOR:

- ▶ Click on the **Workspace** node
- ▶ Select the **Everywhere Server** property group
- ▶ Select, from the **Authorization level** drop-down list, the authorization level that is to have access to Everywhere Server.



#### Information

*Authorization level 0 means write and read access for all users.*

You can find further information in the user administration manual in the Authorization levels chapter.

#### INI ENTRY:

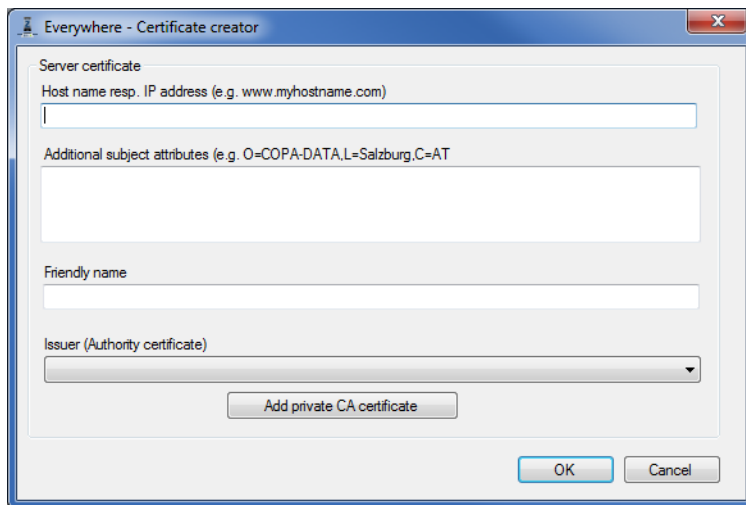
To configure the level of user authorization for Everywhere Server, the following must be added to the zenon6.ini file:

Entry	Description
[PASSWORD]	Configuration of the access authorization for Everywhere Server.
EVWH_ACCESS=	Authorization level that has access to the Everywhere Server. Value: Number of the configured authorization level  Default: 0 (= everyone has access)

### 5.1.3 Everywhere - Certificate creator

The **Everywhere - Certificate creator** is used to create new server certificates. Certificates are required for secure communication between the client and server. You can find this program in the folder C:\Program Files (x86)\Common Files\COPA-DATA\STARTUP.

To start the program, double-click on `Everywhere.CertificateCreator.exe`. A configuration dialog is opened:



Parameters	Description
<b>Server certificate</b>	
<b>Host name resp. IP address</b>	<p>Name or IP address via which the Everywhere Server is accessed by the clients.</p> <p>This is either the IP address or the name of the computer on which the service is running, or the address/name of the firewall/router that connects the computer to the internet.</p>
<b>Additional subject attributes</b> (e.g. O=COPA-DATA, L=Salzburg, C=AT)	Additional attributes for the certificate (optional).
<b>Friendly name</b>	<p>Name for the display in the program <b>Everywhere.Config.exe</b>.</p>
<b>Issuer (Authority certificate)</b>	<p>Issuer certificate or certificate that is to be used to verify the server certificate. The certificates present are shown in the drop-down list.</p> <p>The entry in the drop-down list is based on the <b>Friendly name</b> from the <b>Create root certificate</b> dialog.</p>
<b>Add private CA certificate</b>	<p>Opens dialog to configure the root certificate (on page 15).</p> <p>Note: The root certificate is a mandatory part of the server certificate!</p>
<b>OK</b>	Applies settings and closes the dialog.
<b>Cancel</b>	Discards all changes and closes the dialog.



### Attention

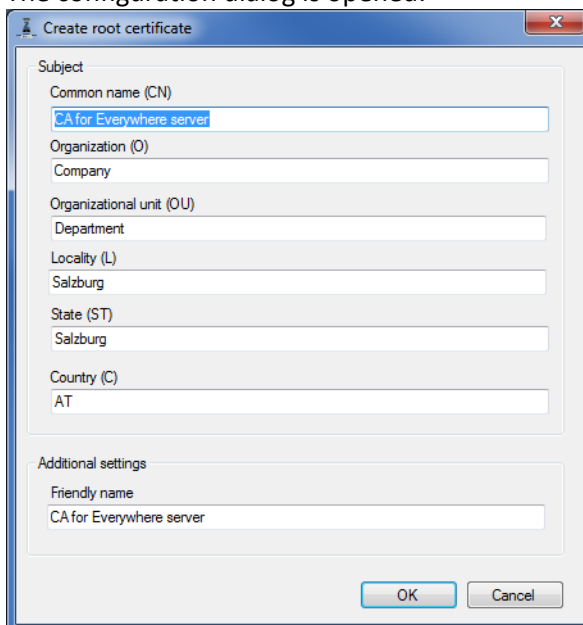
Administrator rights are required to execute this program.

## 5.1.4 Create root certificate

A root certificate is used on the mobile device for secure communication between the server and mobile devices (iPhone, Windows-Phone). This ensures that the end device is also actually connected to the given server.

To create your own root certificate:

- ▶ Click, in the `Everywhere - Certificate creator` (on page 13), the **Add private CA certificate** button.
- ▶ The configuration dialog is opened:



Parameters	Description
<b>Subject</b>	
<b>Common name (CN)</b>	General name Default: CA for Everywhere Server
<b>Organization (O)</b>	Company name Default: Company
<b>Organizational unit (OU)</b>	Name of the organizational unit (department name) Default: Department
<b>Locality (L)</b>	Locality name Default: Salzburg
<b>State (ST)</b>	State or district name Default: Salzburg
<b>Country (C)</b>	Country name Default: AT
<b>Additional settings</b>	Additional information
<b>Friendly name</b>	Short name Default: CA for Everywhere Server
<b>OK</b>	Accepts the changes and opens the save dialog to save the root certificate.
<b>Cancel</b>	Discards all changes and closes the dialog.

### 5.1.5 Certificate on the client

The certificate is checked on the Client. If this certificate has not been created by a root certificate from a generally-known authority (=purchase certificate), the certificate used must be installed on the Client.

#### INSTALLATION OF A CERTIFICATE ON THE CLIENT

Unverified certificates must be installed on the Client in order for them to be accepted.

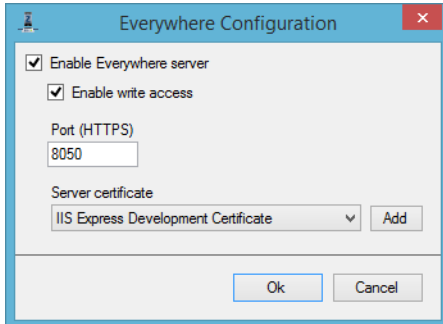
Carry out the following steps for the installation:

1. Place the exported certificate (.cer file) on a web server or FTP server.
2. Open this certificate on the client using the browser.
  - Alternatively, it can also be sent as an email attachment and opened on the client.



## 5.1.6 Certificates

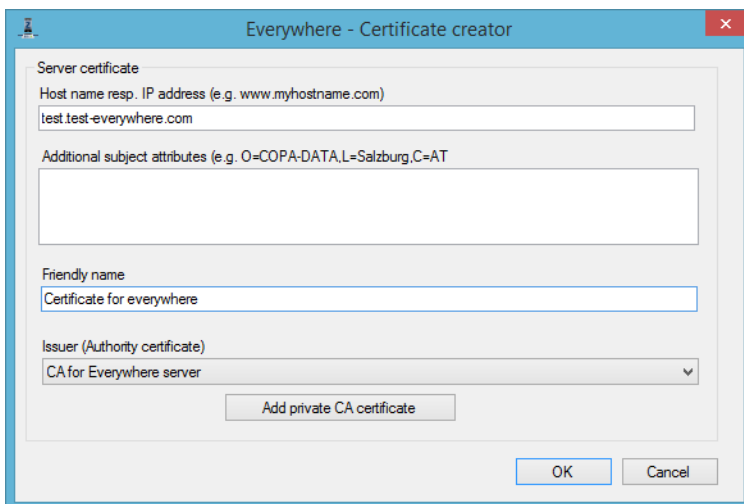
To install Everywhere Server by zenon correctly, carry out the following steps:



1. Start the `Everywhere.Config.exe` program
2. Activate the **Enable Everywhere server** checkbox
3. Enter the correct port in the **Port (HTTPS)** input field.
4. Select a server certificate from the **Server certificate** drop-down list.

### CREATE A SERVER CERTIFICATE

To create a server certificate:



1. Click the **Add** button in the `Everywhere.Config.exe` program to start the `Everywhere - Certificate creator` program.
2. Enter your data into the input fields.
3. Assign a **Friendly name**.
4. Select an SLL certificate from the **Issuer (Authority certificate)** drop-down list.

## CREATION OF A ROOT CERTIFICATE (SLL CERTIFICATE)

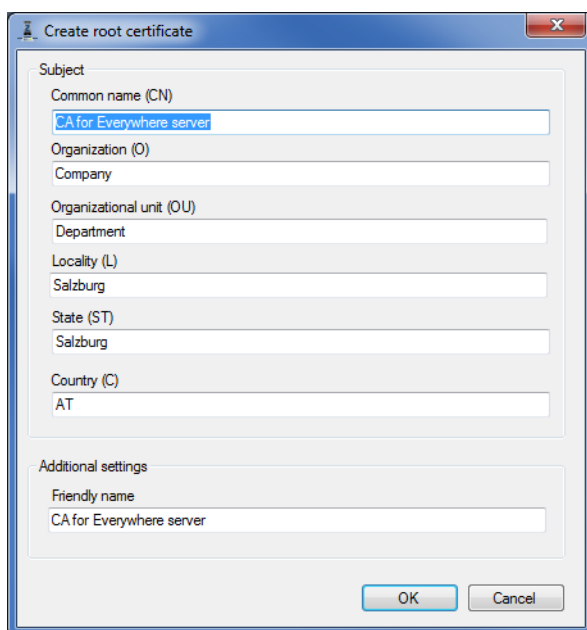
Purchase a certificate from a provider of SLL certificates.

Installation on the Client becomes superfluous as a result, because common certificate providers are already installed on the end device by default.

Search for "SLL certificates" on the Internet. Some providers are, for example:

1. GoDaddy (<http://www.godaddy.com>)
2. VeriSign (<http://www.verisign.com>)
3. CAcert (<http://wiki.cacert.org/FAQ/AboutUs>)

Alternatively, create your own root certificate:



1. Click, in the Everywhere – Certificate creator, the **Add private CA certificate** button.
2. Enter your data into the input fields.
3. Assign a **Friendly name**.
4. Click on the **OK** button
5. The dialog to save the SLL certificate appears.
6. Name your SLL certificate and save it.



### Information

This saved root certificate is the certificate that must be installed on the Client.

## 5.2 CEL entries

When logging into the Everywhere Server, a CEL entry is created if

- ▶ the login data sent was checked by Runtime.

This happens:

- On the first request of a client.
- Every 5 minutes
- Always before a write request if more than a second has passed since the last check.

For variables that have not been acknowledged, alarms and CEL entries are supported by an incremental transfer of value changes.

**LOG ENTRIES:**

Level error	Description
<b>Level Error</b>	Errors that occur during the execution of the Everywhere Server.  However not an error that has something to do with the execution of client requests.
<b>Level Warning</b>	Errors that occur during the execution of the Everywhere Server.  However not an error that has something to do with the execution of client requests.
<b>Level Success</b>	Successfully-executed client requests that lead to a change of the internal status of the Everywhere Server or the zenon Runtime.
<b>Level Failed</b>	<ul style="list-style-type: none"> <li>▶ Error in the execution of client requests</li> <li>▶ Error in the checking of credentials</li> </ul>
<b>Level Msg</b>	Everywhere Server status: <ul style="list-style-type: none"> <li>▶ Start / stop</li> <li>▶ Successful creation of sessions</li> <li>▶ Ending of sessions</li> </ul>
<b>Level Debug</b>	Incoming and outgoing requests with <ul style="list-style-type: none"> <li>▶ Session-ID,</li> <li>▶ URI (Uniform Resource Identifier)</li> <li>▶ Method and status</li> </ul>
<b>Level Deep Debug</b>	Status of the Everywhere Server as well as incoming and outgoing requests, compiled into an entry (Level Msg + Level Debug).

## 5.3 Data model of the Everywhere Server

The **Open Data Protocol (= OData)** is used as an interface for the inclusion of data in your own web application. This protocol is a "WCF data service".

This data service is a "**RESTful services**" (**RE**presentational **S**tate **T**ransfer **A**rchitecture – **REST**). Only methods for manipulating or querying the data model are defined in this (**CRUD** – **C**reate – **R**ead – **U**ppdate – **D**elele).

Because these methods cover the HTTP commands ((**GET**, **PUT**, **POST**, **DELETE**)) data services can be implemented very well via the HTTP protocol. In doing so, the desired object is identified by means of the URL and the operation via the respective HTTP command.



#### Hint

*PUT, CREATE and DELETE must "point" to different instances.  
You can find further information in the official ODATA online documentation.  
Link: <http://www.odata.org/documentation/odata-version-2-0/operations/>  
(<http://www.odata.org/documentation/odata-version-2-0/operations/>)*

The data is, like a relational database, organized in tables. Filter criteria can be defined for querying data (comparable to an SQL query).

**Note:** In contrast to traditional data services, a session-dependent data model is used in the Everywhere service. This means that the data model can - depending on the session - contain different data.



#### Information

The Everywhere service is a WCF data service.  
You can find details and comprehensive documentation for the OData services at:  
[www.odata.org](http://www.odata.org) (<http://www.odata.org>).

### 5.3.1 Data model - static project data

#### EQUIPMENT MODELS (COLLECTION)

- ▶ Type: Everywhere.Service.Data.ZenonEquipmentModel
  - Name (string)
  - Guid (string, key)
  - EquipmentGroups (collection, Everywhere.Service.Data.ZenonEquipmentGroup)

#### EQUIPMENTGROUPS (COLLECTION)

- ▶ Type: Everywhere.Service.Data.ZenonEquipmentGroup

- Guid (string, key)
- Name (string)
- ParentGroup (**Everywhere.Service.Data.ZenonEquipmentGroup**)
- ChildGroups (**collection, Everywhere.Service.Data.ZenonEquipmentGroup**)
- Model (Everywhere.Service.Data.ZenonEquipmentModel)

#### **VARIABLES (COLLECTION)**

- ▶ Type: Everywhere.Service.Data.ZenonEquipmentModel
  - EquipmentGroupGuids (String, commaseparated GUIDs of the related groups)
  - MinimumValue (Real)
  - MaximumValue (Real)
  - Unit (String)
  - ResourcesLabel (String)
  - Identification (String)
  - Name (String)
  - Id (int, key)

#### **ALARMCLASSES (COLLECTION)**

- ▶ Type: Everywhere.Service.Data.ZenonAlarmClass
  - EquipmentGroupGuids (String, commaseparated GUIDs of the related groups)
  - Description (String)
  - Number (int, key)
  - Name (String)

#### **ALARMGROUPS (COLLECTION)**

- ▶ Type: Everywhere.Service.Data.ZenonAlarmGroup
  - EquipmentGroupGuids (String, commaseparated GUIDs of the related groups)
  - Description (String)
  - Number (int, key)

- `Name (String)`

#### ARCHIVES (COLLECTION)

- ▶ Type: `Everywhere.Service.Data.ZenonArchive`
  - `Id (String, key)`
  - `Name (String)`
  - `Values (Collection, Everywhere.Service.Data.ZenonArchiveValue)`

#### ARCHIVEVALUES (COLLECTION)

- ▶ Type: `Everywhere.Service.Data.ZenonArchiveValue`
  - `ArchiveId (String, key)`
  - `VariableName (String)`
  - `Timestamp (Int64, Unixtime [ms], key)`
  - `Type (int, key, None = 0, Sum = 1, Average = 2, Min = 3, Max = 4)`
  - `Value (string)`
  - `VariableId (int, key)`

### 5.3.2 Data model - project data that changes

Below you can find the data model for values that update themselves.

#### ALARMS (COLLECTION)

- ▶ Type: `Everywhere.Service.Data.ZenonAlarm`
  - `Deleted (bool)`
  - `Acknowledged (bool, writeable)`
  - `VariableStatus (Int64)`
  - `Value (string)`
  - `AlarmClass (Everywhere.Service.Data.ZenonAlarmClass)`
  - `AlarmGroup (Everywhere.Service.Data.ZenonAlarmGroup)`
  - `UserId (String)`

- `ReactivateNumber (int)`
- `ReactivateStatus (Int64)`
- `ReactivateTime (Int64, Unixtime [ms])`
- `TimeAcknowledged (Int64, Unixtime [ms])`
- `TimeCleared (Int64, Unixtime [ms])`
- `TimeReceived (Int64, Unixtime [ms], key)`
- `ComputerName (string)`
- `Comment (string, writeable)`
- `Text (string)`
- `Variable (Everywhere.Service.Data.ZenonVariable)`
- `AlarmStatus (int)`
- `Cookie (int)`
- `Id (int, key)`



### Example

The "[Acknowledged](#)" and "[Comment](#)" properties can be used to acknowledge an alarm in zenon or to save a comment.

These properties can then be transferred to the server with the "[PUT](#)" or "[MERGE](#)" command.

## CEL (COLLECTION)

- Type: `Everywhere.Service.Data.ZenonEvent`
  - `VariableStatus (Int64)`
  - `Value (string)`
  - `AlarmClass (Everywhere.Service.Data.ZenonAlarmClass)`
  - `AlarmGroup (Everywhere.Service.Data.ZenonAlarmGroup)`
  - `UserId (String)`
  - `TimeReceived (Int64, Unixtime [ms], key)`
  - `ComputerName (string)`
  - `Comment (string )`
  - `Text (string)`



- Variable (**Everywhere.Service.Data.ZenonVariable**)
- Cookie (int)
- Id (int, key)

### ONLINEVARIABLES (COLLECTION)

- ▶ Type: Everywhere.Service.Data.ZenonOnlineVariable
  - Id (int, key)
  - Variable (**Everywhere.Service.Data.ZenonVariable**)
  - Status (Int64)
  - Timestamp TimeReceived (Int64, Unixtime [ms])
  - Value (string, writeable)



#### Example

- ▶ To sign variables in again, a new instance must be created in the collection. This is carried out with the **CREATE** command. This registration is removed again with **DELETE**.
- ▶ For identification of the online variable that was created on the client, its ID must be established. This ID corresponds to the ID of the zenon variable whose values are to be queried.  
**PUT** or **MERGE** writes a value to a variable.

### 5.3.3 Querying archives

Archive variables can only be queried from the respective project of the Everywhere Server. Cross-project variables are not supported.

#### NOTE ON THE QUERYING OF ARCHIVE VALUES:

In order to query archive data, the following properties must also be entered for the server query:

- ▶ Archive ID
- ▶ Start time
- ▶ End time



### Attention

Values without these filter criteria being stated cannot be queried.

#### Examples:

- ▶ [https://89.26.13.1:8050/zendata/DEMO\\_AUTOMOTIVE/ArchiveValues?\\$format=json&\\$filter=ArchivedId%20eq%20%27FI%27%20and%20Timestamp%20gt%201377690800000L%20and%20Timestamp%20lt%201377690862000L](https://89.26.13.1:8050/zendata/DEMO_AUTOMOTIVE/ArchiveValues?$format=json&$filter=ArchivedId%20eq%20%27FI%27%20and%20Timestamp%20gt%201377690800000L%20and%20Timestamp%20lt%201377690862000L)
- ▶ [https://89.26.13.1:8050/zendata/DEMO\\_AUTOMOTIVE/Archives%28%27FI%27%29/Values?\\$format=json&\\$filter=Timestamp%20gt%201379934601375L%20and%20Timestamp%20lt%201379934601377L%20and%20VariableName%20eq%20%27BodyShop1.Actual%27](https://89.26.13.1:8050/zendata/DEMO_AUTOMOTIVE/Archives%28%27FI%27%29/Values?$format=json&$filter=Timestamp%20gt%201379934601375L%20and%20Timestamp%20lt%201379934601377L%20and%20VariableName%20eq%20%27BodyShop1.Actual%27)

### 5.3.4 Calling up the data model using a web browser

The data model of the Everywhere Server can be displayed in a web browser. This can be used to give you a quick overview and test the first programming lines.

To call up a data model, use Mozilla Firefox or Google Chrome (recommended).

**Attention:** Microsoft Internet Explorer does not support the full functionality!

#### URI ADDRESS

**https://server address or DNS name:port number\zendata\project name**

Parameters	Description
<b>https://</b>	URI schematic of the WWW communication protocol. https: stands for the secure transfer protocol Hypertext Transfer Protocol Secure
<b>Server address or DNS name</b>	IP address or DNS name of the Everywhere Server.
<b>:</b>	<b>:</b> (colon) Separates the server address from the port number.
<b>Port number</b>	Port number for communication, as configured in Everywhere Server (on page 10).
<b>zendata</b>	Constant: is always zendata
<b>Project name</b>	Name of the zenon project configuration.



### Example

[https://89.26.13.1:8050/zendata/Demo\\_Automotive](https://89.26.13.1:8050/zendata/Demo_Automotive)

## Logging in to the data model

If it is not possible to log in when in secure mode (https://), a warning sign appears in the browser. This warning sign can be skipped by clicking on the respective button (Chrome: "Continue anyway", Firefox: "I am aware of the risk"). Communication then takes place unsecured.

### THE LOGIN DIALOG APPEARS:

#### USER NAME:

The user name is the session ID at the same time. This session ID comprises:

**Time Stamp|Device ID|User Name**

Parameters	Description
<b>Time stamp</b>	32-bit time stamp of the session: <ul style="list-style-type: none"> <li>▶ UTC Unix time (UTF8 coded: 8-digit HEX string)</li> <li>▶ must be 8 digits</li> </ul>
	Pipe character <b>Alt GR</b> key and <-key
<b>Device ID</b>	ID of the end device (mobile device, ...) - normally UUID. The device ID is device-dependent and must be unique.
	Pipe character <b>Alt GR</b> key and <-key
<b>User name</b>	User name, as configured in zenon. When entering this, note capitalization. <b>Note:</b> The user name must not contain a colon.



### Example

*538497B4|550e8400-e29b-11d4-a716-446655440000|Manager*

#### PASSWORD:

The password comprises:

#### Type of password sending and password

Parameters	Description
Type of password sending	<p>How the password is sent:</p> <ul style="list-style-type: none"> <li>▶ = (equal sign) unencrypted</li> <li>▶ + (plus sign) encrypted</li> </ul> <p><b>Note:</b> The password must be preceded with the type of sending (= or +). Otherwise logging in to the data model will not be successful!</p>
Password	<p>Password as configured in zenon.</p> <p><b>Note:</b> Pay attention to capital letters and small letters when entering the data!</p>



### Example

*+Manager*

*Encrypted communication with the Everywhere Server - login with the password*

*Manager.*

## 5.3.5 Examples

You can find some examples of code here. These examples should show how values from the server can be updated.

**Note:** The examples use the Java programming language.

Example of code for logging in to an Everywhere Server:

```
// get authentication information from the user interface
```

```

String userName = arg0[0].getString("UserName");
String password = arg0[0].getString("Password");
String server = arg0[0].getString("Server");
String project = arg0[0].getString("Project");
String port = arg0[0].getString("Port");
Date curTime = new Date();

// full username -> as described
String fullUserName = Long.toHexString(curTime.getTime())+"|";           // 8
digit hex timestamp
fullUserName = fullUserName + Settings.Secure.ANDROID_ID+"|";
fullUserName = fullUserName + userName;

// set the authentication to basic https (OClientBehaviour from odata4j)
OClientBehavior authBehavior = OClientBehaviors.basicAuth(fullUserName, "="
+ password);

OClientBehavior allowSelfSignedBehavior =
AllowSelfSignedCertsBehavior.allowSelfSignedCerts();

// Consumer declaration: private ODataConsumer consumer;
// here the url to the server has to be inserted
// we have https:"the server IP or name":Portnumber(for HTTPS
8050)/zendata(fixed)/Projectname
consumer =
ODataConsumers.newBuilder("https://"+server+": "+port+"/zendata/"+project).setClientBeh
aviors
(authBehavior, allowSelfSignedBehavior).build();

```

## Examples for browser display

A browser can be used to display your zenon project configuration in a browser.

Note: The web addresses shown here relate to the demo project of the Everywhere Server.

## DATA MODEL - OVERVIEW

To display a simplified overview of your data model in your browser, enter the following URI into the browser:

[https://89.26.13.1:8050/zenodata/DEMO\\_AUTOMOTIVE](https://89.26.13.1:8050/zenodata/DEMO_AUTOMOTIVE)

## DATA MODEL - DETAILED OVERVIEW

The following URI lists the complete data model:

[https://89.26.13.1:8050/zenodata/DEMO\\_AUTOMOTIVE/\\$metadata](https://89.26.13.1:8050/zenodata/DEMO_AUTOMOTIVE/$metadata)

## DISPLAY OF VARIABLES

You get an overview of all variables with the URI:

[https://89.26.13.1:8050/zenodata/DEMO\\_AUTOMOTIVE/Variables?\\$format=json](https://89.26.13.1:8050/zenodata/DEMO_AUTOMOTIVE/Variables?$format=json)

## DISPLAY OF VARIABLES WITH FILTERING

The display of variables can be filtered. The address below displays the variable with the ID 321

[https://89.26.13.1:8050/zenodata/DEMO\\_AUTOMOTIVE/Variables?\\$format=json&\\$filter=Id%20eq%20321](https://89.26.13.1:8050/zenodata/DEMO_AUTOMOTIVE/Variables?$format=json&$filter=Id%20eq%20321)

**Note:** You can find a precise description of all filters in the OData online documentation. You can find this documentation on the Internet at the address [www.odata.org](http://www.odata.org) (<http://www.odata.org>).

## RECEIVING ALARMS

The example shows how alarms are updated. This syntax can also be used for the other data types in the same way.

```
private class RequestPollAlarmsTask extends ConnectionTask{

    public RequestPollAlarmsTask(EverywhereConnectionHandler handler) {
        super(handler);
    }

    @Override
    protected void doInBackground(Bundle... arg0) {

        try {
            lock.acquire();
        } catch (InterruptedException e1) {
            // TODO Auto-generated catch block
        }
    }
}
```

```

        e1.printStackTrace();
    }

    int updateIntervall = arg0[0].getInt("Intervall");
    String filter = arg0[0].getString("Filter");

    try
    {
        // Consumer declaration: private ODataConsumer consumer;
        // alarms declaration: private Enumerable <OEntity>
        alarms =
consumer.getEntities("Alarms").expand("Variable").execute();
        isConnected=true;
    }
    catch (Exception e)
    {
        isConnected=false;
        connectionException=e;
    }
    lock.release();
    return null;
}

```

### Example of code - calling up OEntity objects from the server

This example shows how information in relation to an OEntity object can be called up from the server. This data can thus be taken on for the user programming. This example also shows how properties can be called up from a data object.

```

/**
 *
 * @param position: Object at which position of the entityset should be returned
 * @return the finished Item Alarm for the Listt of Alarms
 */
protected Alarm GetAlarmFromEntity(int position) {

    Alarm retAlarm = new Alarm();

    // We use position as listItemId. Here we could use some other unique id

```

```

        // to reference the list data
        ListView listViewAlarms = (ListView) findViewById(R.id.Alarmlist);
        OEntity var = alarms.elementAt(position).getLink("Variable",
OLink.class).getRelatedEntity();

        int EntityItemCnt= alarms.count();
        OProperty<String> varName = var.getProperty("Name", String.class);

        OProperty<String> alarmText =
alarms.elementAt(position).getProperty("Text", String.class);
        String Header = varName.getValue() + " - " + alarmText.getValue();

        // Body data
        Bundle bodyBundle = new Bundle();
        OProperty<?> alarmValue =
alarms.elementAt(position).getProperty("Value");
        String timeReceived =
alarms.elementAt(position).getProperty("TimeReceived").getValue().toString();
        String timeCleared =
alarms.elementAt(position).getProperty("TimeCleared").getValue().toString();
        String Acknowledged =
alarms.elementAt(position).getProperty("Acknowledged").getValue().toString();
        Date date = new java.util.Date(Long.parseLong(timeReceived));
        String dateString = new SimpleDateFormat("dd.MM.yyyy HH:mm").format(date);

        retAlarm.setAlarmName(varName.getValue());
        retAlarm.setTimeReceived(dateString);
        retAlarm.setValue(alarmValue.getValue().toString());
        retAlarm.setAcknowledged(Acknowledged);
        retAlarm.setCleared(timeCleared);
        retAlarm.setAlmText(alarmText.getValue());

        return retAlarm;
    }

```

## Other examples of code

Other examples for programming with OData are on the Internet.



The example given shows how information relating to an OEntity objects can be called up from the server to the end device. This example also shows how properties of a data object can be received.

Link: <https://github.com/vorburger/odata4j/tree/master/odata4j-core/src/test/java/org/odata4j>  
(<https://github.com/vorburger/odata4j/tree/master/odata4j-core/src/test/java/org/odata4j>)



#### Attention

*This Internet site is not shown in Internet Explorer!*

### 5.3.6 Using the app on a mobile device (Android)

In order to use your own applications on a mobile end device (Android), compile it with the **Eclipse** program.

**Eclipse** is a programming tool for developing software and was originally used as an integrated development environment for the Java programming language.



#### Information

**Eclipse** is available as a free download.

#### INSTALLATION:

In order to be able to use **Eclipse** for your development environment, install the program including the Android Developer Bundle.

The following components must be installed on the development computer:

- **The Android Developer Bundle**

The Android Developer Bundle is available free on the Internet.

Link to the installer: <http://developer.android.com/sdk/index.html>  
(<http://developer.android.com/sdk/index.html>)

It is important that the correct driver for the mobile device is installed on the Desktop computer. Otherwise no communication between the development computer and the end device is possible.

- **The Java Development and Runtime Environment**

The Java Development Environment (JRE) and the Java Runtime Environment (JSE) must be installed on the development computer.

JRE and JSE can be downloaded directly from the Java website. Both environments must be installed separately.

Link to download: [www.oracle.com/technetwork/java/javase/downloads/index.html](http://www.oracle.com/technetwork/java/javase/downloads/index.html)  
(<http://www.oracle.com/technetwork/java/javase/downloads/index.html>)

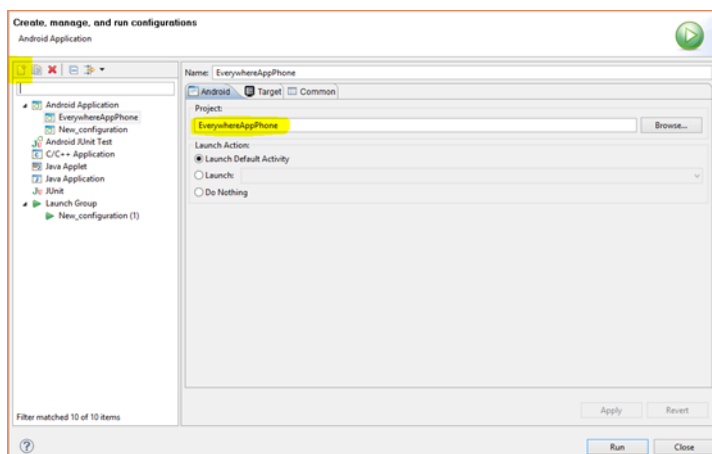
► **Required libraries:**

To be able to use OData in **Eclipse**, you then need to link the appropriate library for your preferred development environment in **Eclipse**.

Libraries for the common programming languages can be found on the OData Internet site. Select your desired programming language in the tabs shown in order to get to the respective downloads.

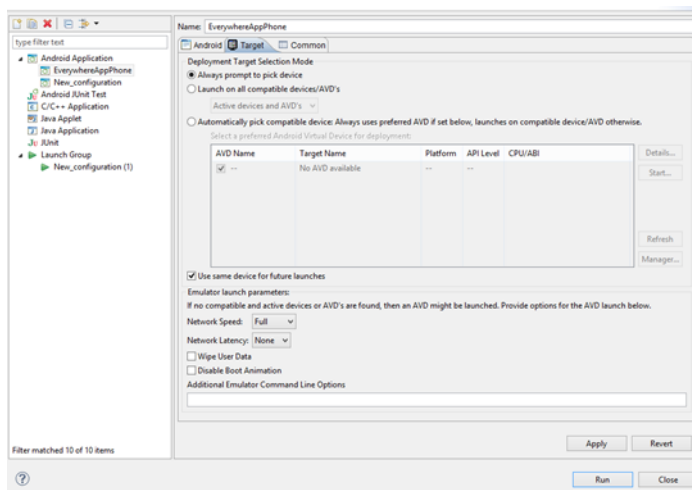
Link to the libraries: [www.odata.org/libraries](http://www.odata.org/libraries) (<http://www.odata.org/libraries>).


## Recommended settings for the Eclipse programming environment



To configure **Eclipse** for mobile end devices, carry out the following steps:

1. Select, the **Run configurations...** entry in the **Run** menu bar.
2. The configuration dialog for a new **Run** configuration appears.
3. Click on the symbol **New launch configuration**.
4. In the **Android Application** tree, the new entry **NEW\_configuration** is added.
5. Name your Android app project in the **Android** tab.
6. Click on the **Browse...** button to select your Android project. This is then transferred to the mobile device.
7. Now connect your mobile end device to the development computer.



8.  Configure your target device in the **Target** tab.
9. Ensure that the **Always prompt to pick device** property is activated.  
**Note:** In this case, you are asked before execution which device your app is to be executed on.
10. Start the debug process:
11. Start the app.  
**Note:** For the remaining entries, it is recommended that the existing default settings are left unchanged.
12. After clicking on the **Play** or **Debug** buttons, you are requested to select your end device or your emulator in order to start the program.
13. The app is then started automatically on the end device.

## 6. Everywhere App by zenon

Apps allow the visualization of a zenon project on mobile end devices, tablets and desktop computers. The range of functions is identical. The graphical user interface varies depending on the operating system used.

The respective apps guide you through your project configuration step by step. You select the respective parameters you want and thus receive the attendant information. Navigation is intuitive and follows a chronological sequence.

## 7. The following Apps are available:

- ▶ Everywhere App by zenon for MS Windows Phone (on page 38)
- ▶ Everywhere App by zenon for iPhone and iPad (on page 65)
- ▶ Everywhere App for Desktop for Desktop (on page 85)
- ▶ Everywhere App by zenon (on page 97) for Android



### License information

*All apps are available in the manufacturers' stores free of charge.*

### 7.1 Requirements

The Everywhere App by zenon visualizes the project configuration provided by Everywhere Server (on page 7) on a smartphone, tablet or desktop computer.

#### SYSTEM REQUIREMENTS

- ▶ Windows Phone 7.5 or higher (Windows Phone 8 or higher recommended)
- ▶ iOS 6 or higher for Apple iPhone and iPad
- ▶ Windows 8 or higher for Everywhere App for Desktop
- ▶ Android 2.2.x - Froyo or higher



### Information

*The Everywhere App by zenon user interface is only available in English.*

### 7.2 Installation

Carry out the following steps in order to be able to use Everywhere App by zenon on your device:

- ▶ Start the app store.
- ▶ Enter `zenon` or `Everywhere App` as a search term.
- ▶ Select Everywhere App by zenon.



#### **Hint**

A profile for a demo project is already installed during the installation. You can thus also test Everywhere App by zenon if you have not yet configured your own project with zenon.

### 7.2.1 Note for OEM user:



#### **Information**

Always select Everywhere App by zenon!  
The app is used for different products.

*Everywhere App by zenon is compatible with all OEM project configurations.*

## **7.3      Everywhere App by zenon for MS Windows Phone**

### **7.3.1      Authentication**

The login screen allows you to log in to an Everywhere Server by zenon and to select user profiles.



# Everywhere

App by zenon

Profile

COPA-DATA demo server

User name

Manager

Password

● ● ● ● ● ● ● ●



Parameters	Description
<b>Profile</b>	Name of the selected profile. <ul style="list-style-type: none"> <li>▶ Tapping on the profile name opens the dialog to select an existing profile.</li> <li>▶ Tapping on the profile name also opens the dialog to create a new profile.</li> </ul>
<b>User name</b>	User name for login at the Everywhere Server.
<b>Password</b>	Password for login at the Everywhere Server.

**Note:** Pay attention to capital letters and small letters when entering the data!

#### APP BAR

<b>login</b> (Symbol >)	Logs the app into Everywhere Server and switches to the next screen.
...	Extends the display of the app bar.



### 7.3.2 Profile List

The profile overview lists existing profiles and administers these.



9:53

 SELECT PROFILE

## COPA-DATA demo server

Server: ATSZG-WKS062

Port: 8050

Project: DEMO\_AUTOMOTIVE



Parameters	Description
Profile List	<p>Lists existing profiles.</p> <p>The profile that is currently selected is highlighted in color.</p> <p>Tapping on the desired profile activates the profile and switches to the login screen.</p>

#### APP BAR

Parameters	Description
<b>edit</b> (pencil symbol)	Tapping on the symbol switches to the symbol display from <b>edit</b> (pencil symbol) to <b>add</b> (+ symbol).
<b>add</b> (+ Symbol)	Tapping on the <b>edit</b> symbol makes the <b>add</b> symbol available. Another tap opens the screen to configure a profile.
...	Extends the display of the app bar.

#### TO EDIT AN EXISTING PROFILE:

- ▶ Tap the **edit** symbol in the app bar.
- ▶ Tap an existing profile from the list.



### 7.3.3 Edit Profile

In this screen, existing profiles can be changed and new profiles can be configured.



9:54

## EDIT PROFILE

Profile name

COPA-DATA demo server

Server host name

ATSZG-WKS062

Port HTTPS

8050

Project name

DEMO\_AUTOMOTIVE

Display variables by

Name

Remember Password

On



Parameters	Description
<b>Profile name</b>	Name of the profile: Freely selectable
<b>Server host name</b>	Name of the Everywhere Server to which a connection is to be created.
<b>Port HTTPS</b>	Port address for the connection to the Everywhere Server. Default: 8050
<b>Project name</b>	Project name for the connection to Everywhere Server.
<b>Display variables by:</b>	Display of the variables. Select from drop-down list: <ul style="list-style-type: none"> <li>▶ <b>Name</b> Variable display according to name</li> <li>▶ <b>Identification</b> Variable display according to identification</li> <li>▶ <b>Res. Label</b> Variable display according to resources label</li> </ul>
<b>Remember password:</b> (Slider)	Save password for input. The setting is also shown as text: <ul style="list-style-type: none"> <li>▶ <b>OFF:</b> (slider left and bright) Password for login is not saved in the profile settings.</li> <li>▶ <b>On :</b> (slider left and dark) Password is saved in the profile settings.</li> </ul> Default: OFF

#### APP BAR

Parameters	Description
<b>Save</b> (Symbol)	Saves the current settings and returns to the profile overview.
<b>Delete</b> (Recycle Bin symbol)	Deletes the selected profile.
...	Extends the display of the app bar.

### 7.3.4 Equipment Model

This screen shows the configured equipment model.

 CHOOSE VARIABLES

Home

# models

Automotive



Tapping on the equipment model names switches to the next screen with the selection of the equipment groups.



**APP BAR**

<b>continue</b> (Symbol >)	Logs the app into Everywhere Server and switches to the next screen.
<b>logout</b> (X-Symbol)	Ends connection to the server and returns to the login screen.
...	Extends the display of the app bar.

### 7.3.5 Equipment Groups

In this screen, the configured equipment groups are listed and selected.

## CHOOSE VARIABLES

Automotive

# groups

Body-Shop

Paint-Shop

Assembly-Shop

Alarm Messages



Tapping on the equipment model names switches to the next screen with the selection of the allocated variables.

**APP BAR**

<b>continue</b> (Symbol >)	Logs the app into Everywhere Server and switches to the next screen.
<b>logout</b> (X-Symbol)	Ends connection to the server and returns to the login screen.
...	Extends the display of the app bar.

### 7.3.6 Variables selection

In this screen, configured variables from the selected equipment group are listed and selected.



CHOOSE VARIABLES

Automotive→Body-Shop

# variables group

☒ BodyShopAll.Actual

☒ BodyShopAll.Delta

☒ BodyShopAll.Sum

☒ BodyShopAll.Target

☒ BodyShopDelta

☒ LinValueTest

☒ StringTest



Lists all available variables. Select the variables by tapping.

**Note:** All variables can be selected or deselected with the **Extended app bar**.

#### APP BAR

<b>continue</b> (Symbol >)	Logs the app into Everywhere Server and switches to the next screen.
<b>logout</b> (X-Symbol)	Ends connection to the server and returns to the login screen.
...	Extends the display of the app bar.

#### EXTENDED APP BAR

(only visible if the app bar has been extended by tapping on . . .)

<b>select all variables</b>	Activates/deactivates all variables in the list.
<b>clear variables</b>	Shows new screen with detail view of the selected variable(s).  Inactive if no variable is selected.

### 7.3.7 Values view

This screen visualizes the values of the selected variables. There are two views available.





ONLINE VIEW

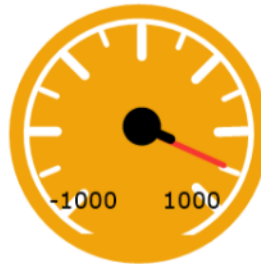
Automotive→Body-Shop

# variables alarm

BodyShopAll.Actual

832.437

1/21/2014, 9:55:52 AM, 640ms  
SPONT, T\_STD



BodyShopAll.Delta

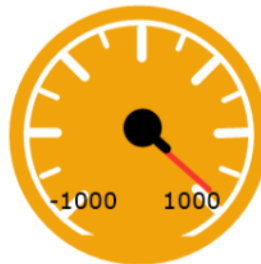
4.437



BodyShopAll.Sum

960

1/21/2014, 9:14:34 AM, 141ms  
SPONT, T\_STD



BodyShopAll.Target

828



**VARIABLE LIST**

Parameters	Description
Variablenliste	<p>List of selected variables:</p> <p>Tapping on the variable switches between "normal view" and "detailed view"</p> <ul style="list-style-type: none"><li>▶ Normal view:<ul style="list-style-type: none"><li>Variable name</li><li>Variable value (numerical)</li><li>Variable value (bar display)</li></ul></li><li>▶ Detail view:<ul style="list-style-type: none"><li>Variable name</li><li>Date and time of last update</li><li>Communication type</li><li>Data type of the variable</li><li>value with pointer instrument display</li></ul></li></ul>



### 7.3.8 Alarm List

This screen visualizes active alarms in a list. Alarms can be acknowledged by tapping on the alarm entry.

## ONLINE VIEW

Automotive→Alarm Messages

# alarms variable

## Not Aus Roboter RS02

### Alarms\_BodyShop[4]

Value: 1  
Received: 1/21/2014, 9:57:13 AM, 50ms  
Cleared: No  
Acknowledged: No

## Not Aus gedrückt

### Alarms\_PaintShop[4]

Value: 1  
Received: 1/21/2014, 9:57:11 AM, 299ms  
Cleared: No  
Acknowledged: No

## Sammelalarm Linie 1

### Alarms\_PaintShop[1]

Value: 1  
Received: 1/21/2014, 9:57:00 AM, 801ms  
Cleared: No



## ALARMS


Parameters	Description
Liste mit Alarmeinträgen	<p>List of active alarms:</p> <ul style="list-style-type: none"><li>▶ Variable name</li><li>Value</li><li>Date and time of the alarm</li><li>Acknowledged alarm: Yes/No</li><li>Cleared alarm: Yes/No</li></ul> <p>Tapping on an alarm entry acknowledges that entry.</p>

## APP BAR

...	Extends the display of the app bar.
-----	-------------------------------------

### 7.3.9 Acknowledge alarms

An alarm can be acknowledged by tapping on the list entry.

 9:57

## Everywhere App

Acknowledge alarm: Sammelalarm Linie 1  
(Alarms\_PaintShop[1])?

Received: 1/21/2014, 9:57:00 AM, 801ms  
Cleared: No  
Acknowledged: No

### Störung Sicherheitsystem Linie 2

#### Alarms\_AssemblyShop[2]

Value: 1  
Received: 1/21/2014, 9:56:57 AM, 301ms  
Cleared: No  
Acknowledged: No

### Störung Schrauber SC01

#### Alarms\_AssemblyShop[3]

Value: 1  
Received: 1/21/2014, 9:56:39 AM, 801ms  
Cleared: No  
Acknowledged: No

...

**ACKNOWLEDGING AN ALARM WITH EVERYWHERE APP:**

1. Tap on the alarm entry in the alarm screen.
2. A dialog is opened.
3. Tap **yes** if you want to acknowledge the alarm.
4. **No** closes the acknowledgment dialog

After acknowledgment, the entry for the alarm (**Acknowledged:**) shows the date and time of acknowledgment.






## **7.4      Everywhere App by zenon for iPhone and iPad**

### **7.4.1      Authentication**

The login screen allows you to log in to an Everywhere Server by zenon and to select user profiles.

Carrier 

5:42 PM



# Everywhere

App by zenon

ATSZG-WKS062.copa-da... 

Manager

● ● ● ● ● ● ●

 Login

The following Apps are available:

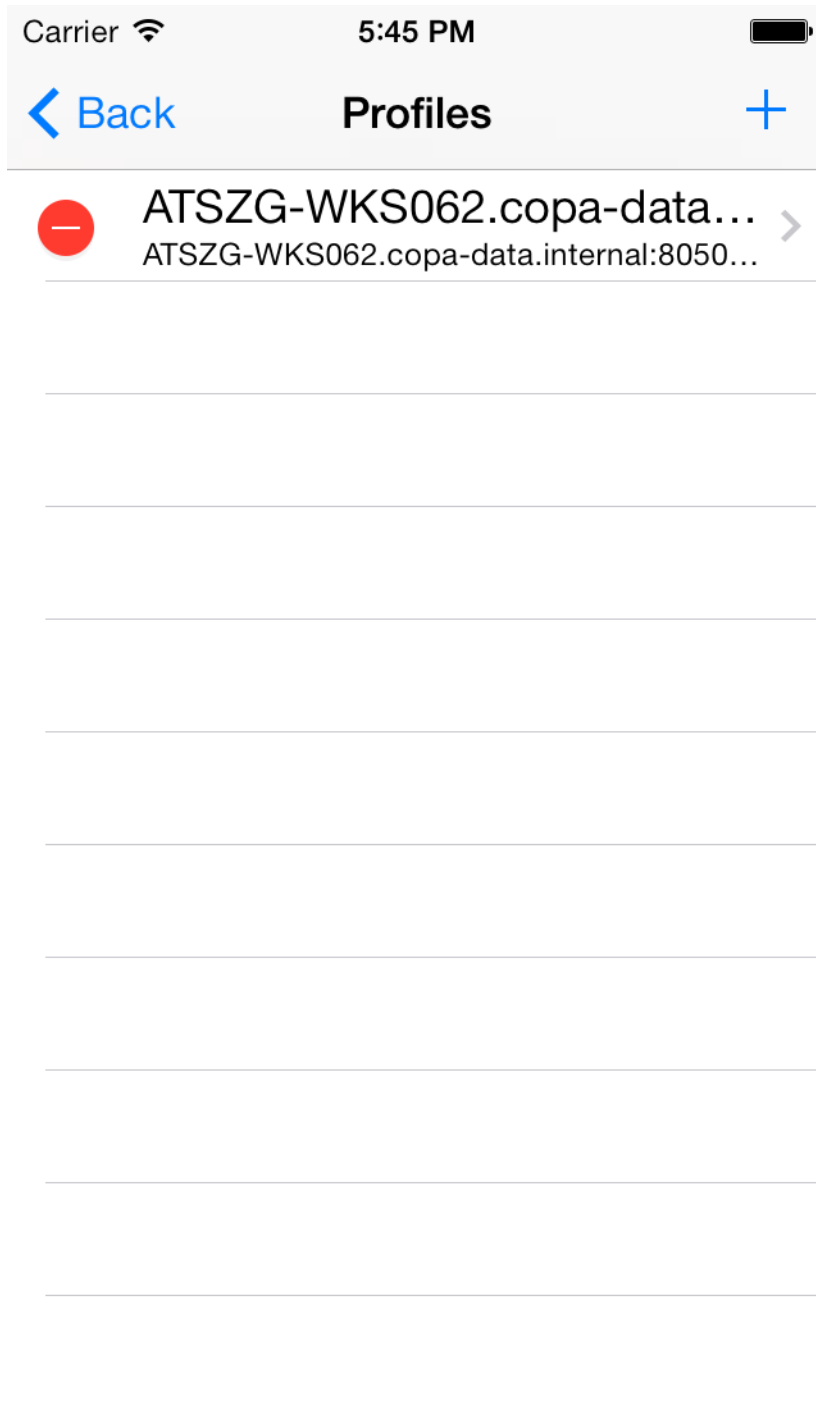


Parameters	Description
<b>Profilname</b>	Name of the selected profile.
(arrow symbol)	Opens the <code>profile overview</code> screen.
<b>Username</b>	User name for login at the Everywhere Server.
<b>Password</b>	Password for login at the Everywhere Server.
<b>Login</b> (Button)	Logs the app into Everywhere Server and switches to the next screen.

**Note:** Pay attention to capital letters and small letters when entering the data!

## 7.4.2 Profile List

The profile overview lists existing profiles and administers these.



Parameters	Description
< Back	Switches back to the login screen
Edit	Switches the list of existing profiles into edit mode  Note: if the <b>Edit</b> button is clicked, it is displayed as <b>+ symbol</b> . A <b>red - in a red cross</b> is placed before it in the list of configured profiles.  (the screen to edit the profiles opens by clicking on the profile name.)
> (Symbol)	Opens the screen to edit the profiles.
Profile name	List of the already-configured profiles.

### CREATING A NEW PROFILE

- To create a new profile, click on the + button


### EDITING A PROFILE

To switch to the editing mode of a profile:


- Tap on the > symbol, on the right next to the profile name.

### 7.4.3 Edit Profile

In this screen, existing profiles can be changed and new profiles can be configured.

Carrier 

5:47 PM



Cancel

Edit profile

Save

Profile name

ATSZG-WKS062.copa-data.internal

Server

ATSZG-WKS062.copa-data.internal

HTTPS Port

8050

Project

DEMO\_AUTOMOTIVE


Display variables by:

Name

Identification

Res. label

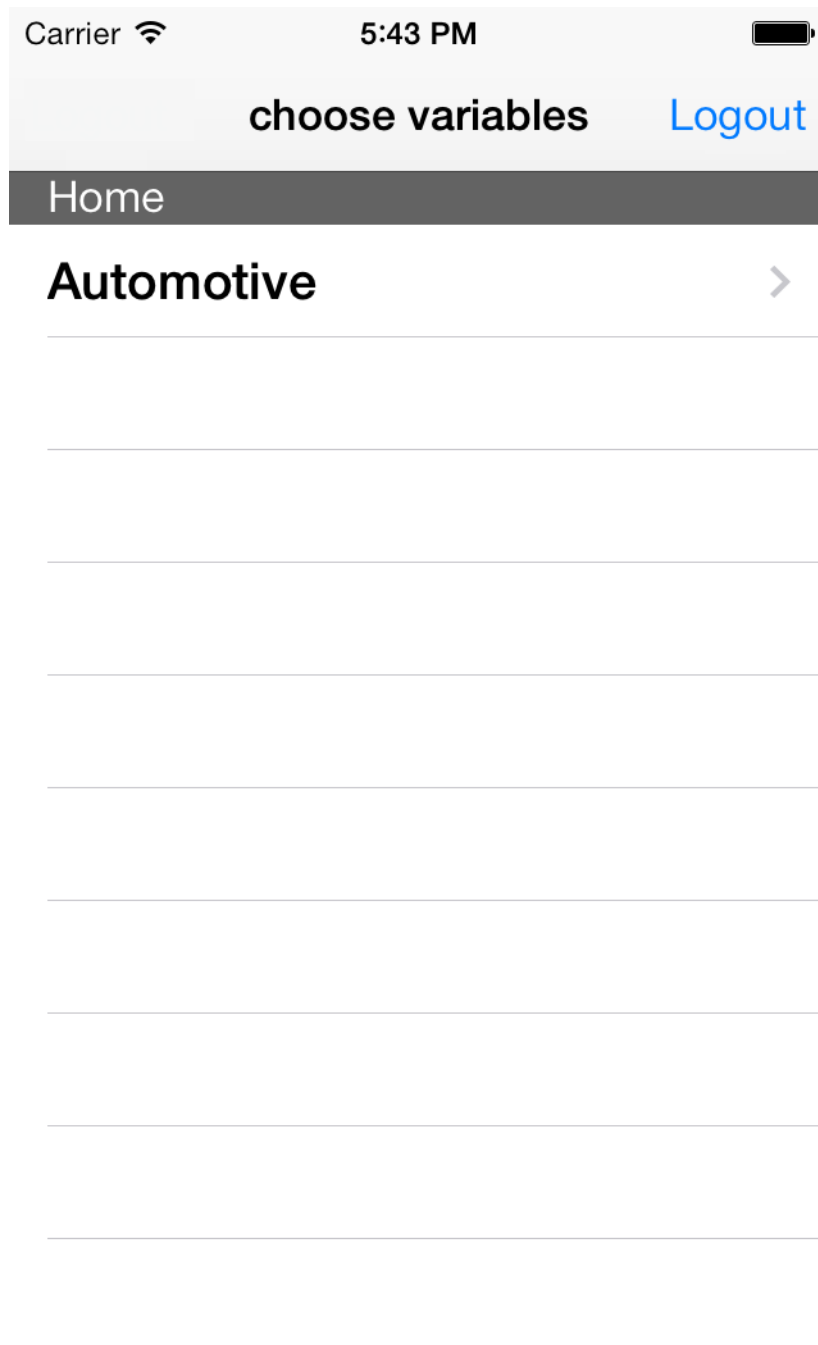
Remember password:



Parameters	Description
<b>Cancel</b>	Cancels input and returns to the profile overview.
<b>Save</b>	Saves the input and returns to the profile overview.
<b>Profile name</b>	Name of the profile: Freely selectable
<b>Server</b>	Name of the Everywhere Server to which a connection is to be created.
<b>HTTPS Port</b>	Port address for the connection to the Everywhere Server.  Default: 8050
<b>Project</b>	Project name for the connection to Everywhere Server.
<b>Display variables by:</b> (Tool bar)	Display of the variables. Select from drop-down list: <ul style="list-style-type: none"> <li>▶ <b>Name</b> Variable display according to name</li> <li>▶ <b>Identification</b> Variable display according to identification</li> <li>▶ <b>Res. Label</b> Variable display according to resources label</li> </ul>
<b>Remember password:</b> (Slider)	Save password for input: <ul style="list-style-type: none"> <li>▶ <b>OFF:</b> (slider left and bright) Password for login is not saved in the profile settings</li> <li>▶ <b>On:</b> (slider right and dark) Password is saved in the profile settings</li> </ul> Default: OFF

#### 7.4.4 Equipment Model

This screen shows the configured equipment model.





The following Apps are available:



Parameters	Description
Logout	Ends connection to the server and returns to the login screen.
Project name	List of available projects on the Everywhere Server
> (Symbol)	Opens the screen to select an equipment model.

### 7.4.5 Equipment groups and variables

This screen lists available equipment groups of the selected equipment model as well as the attendant variables.



**HEADER**

Parameters	Description
<b>Back</b>	Returns to previous screen. Note: Swipe to the right also returns to the previous screen.
<b>Logout</b>	Ends connection to the server and returns to the login screen.

**EQUIPMENT GROUPS**

Parameters	Description
List of equipment groups	
> (Symbol)	

**VARIABLES**

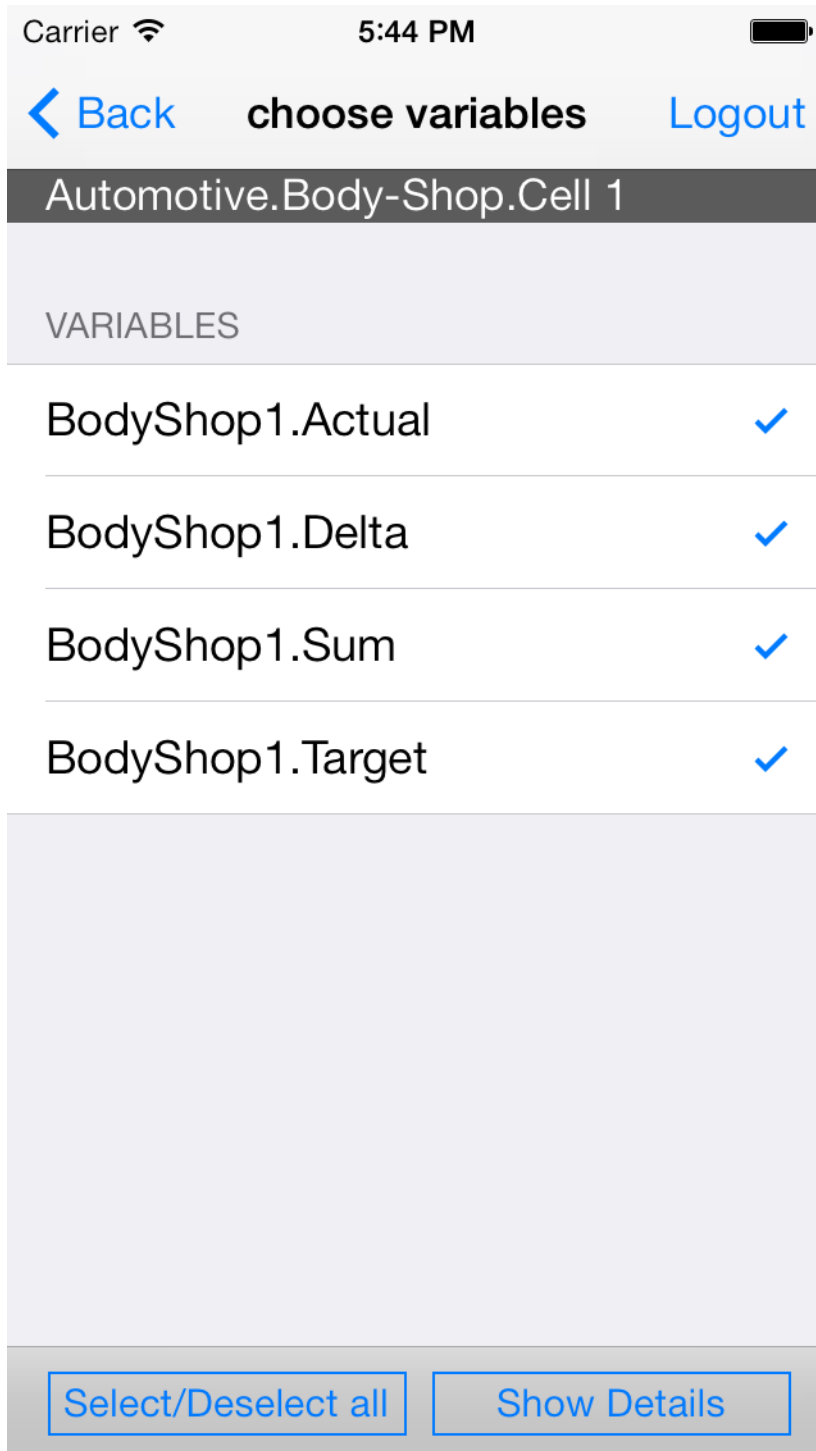
Parameters	Description
Variable list	List of the available variables. Selection/deselection by simply tapping. Selected variables are indicated by a <b>blue tick</b> .

**FOOTER**

Parameters	Description
<b>Select/Deselect all</b>	Activates/deactivates all variables in the list.
<b>Show Details</b>	Shows new screen with detail view of the selected variable(s). Inactive if no variable is selected.

#### 7.4.6 Variables selection

In this screen, configured variables from the selected equipment group are listed and selected.



**HEADER**

Parameters	Description
<b>Back</b>	Returns to previous screen. Note: Swipe to the right also returns to the previous screen.
<b>Logout</b>	Ends connection to the server and returns to the login screen.

**VARIABLES**

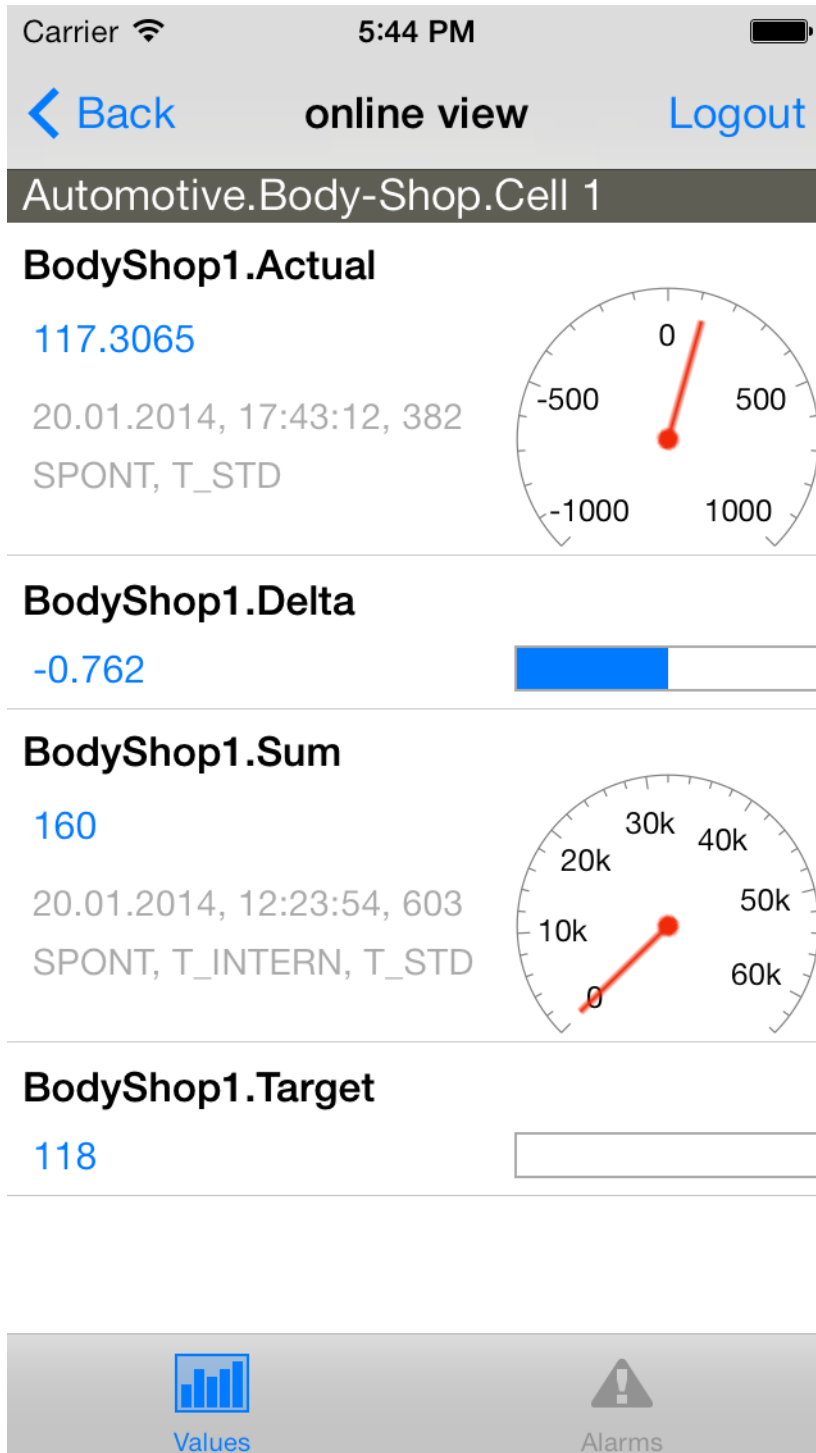
Parameters	Description
Variable list	List of the available variables. Selection/deselection by simply tapping. Selected variables are indicated by a <b>blue tick</b> .

**FOOTER**

Parameters	Description
<b>Select/Deselect all</b>	Activates/deactivates all variables in the list.
<b>Show Details</b>	Shows new screen with detail view of the selected variable(s). Inactive if no variable is selected.

### 7.4.7 Values view

This screen visualizes the values of the selected variables. There are two views available.



**HEADER**

Parameters	Description
<b>Back</b>	Returns to previous screen. Note: Swipe to the right also returns to the previous screen.
<b>Logout</b>	Ends connection to the server and returns to the login screen.

**VARIABLE OVERVIEW**

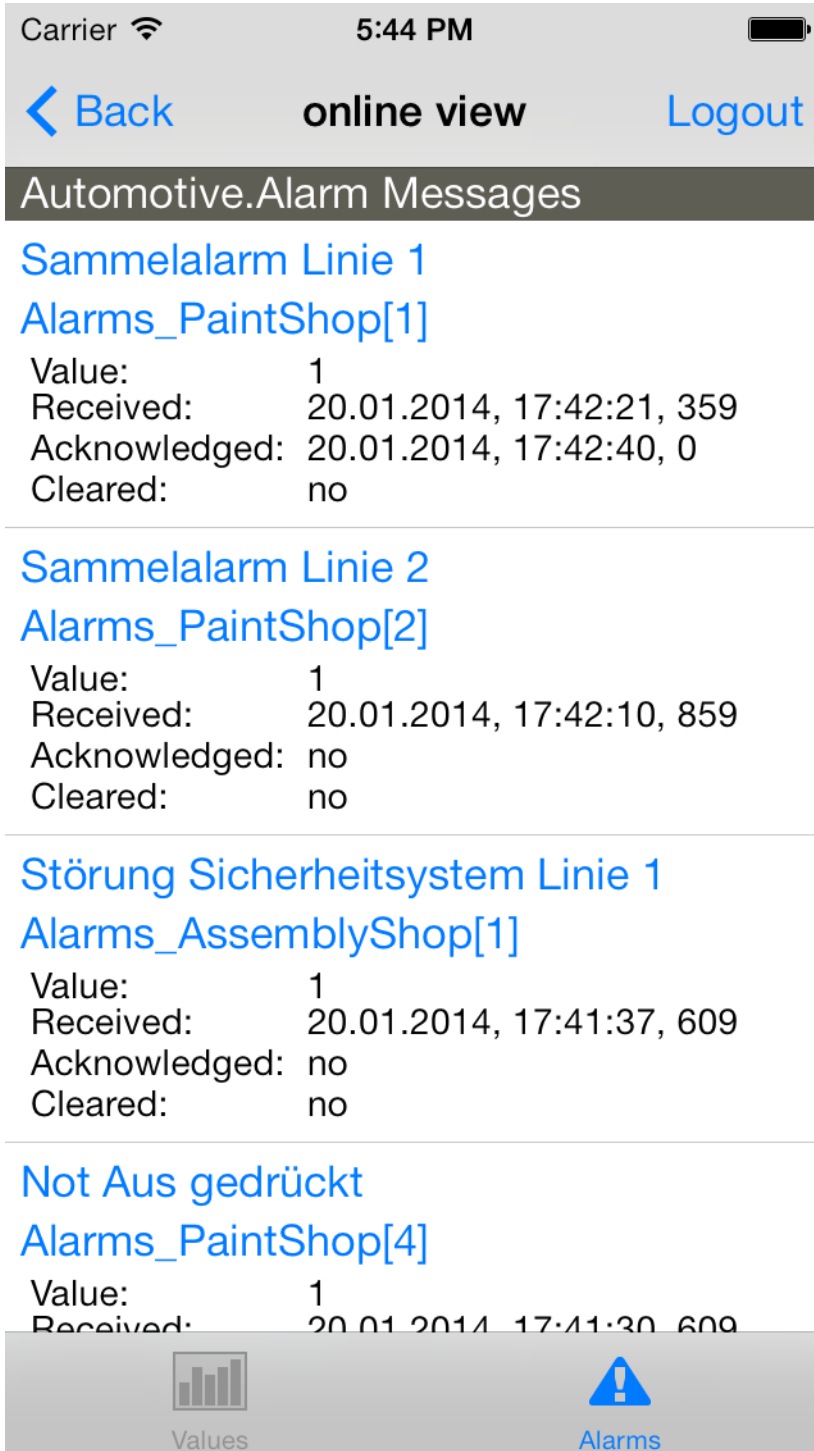
Parameters	Description
Variable list	List of selected variables: <ul style="list-style-type: none"> <li>▶ Normal view: Variable name  Variable value (numerical)  Variable value (bar display) </li> <li>▶ Detail view: Variable name  Date and time of last update  Communication type  Data type of the variable  value with pointer instrument display </li> </ul>

**FOOTER**

Parameters	Description
<b>Values</b> (Symbol)	Switches to display of values of the variables. Blue background if active.
<b>Alarms</b> (Symbol)	Switches to display of alarms. Blue background if active.

### 7.4.8 Alarm List

This screen visualizes active alarms in a list. Alarms can be acknowledged by tapping on the alarm entry.



The screenshot shows the 'Automotive.Alarm Messages' screen in the Zenon application. The status bar at the top indicates 'Carrier' with a signal icon, the time '5:44 PM', and a battery icon. The navigation bar includes a blue back arrow, the text 'online view', and a blue 'Logout' link. The main content area lists four active alarms, each with its name, a link to its details, and a table of status information.

Alarm Name	Value	Received	Acknowledged	Cleared
Sammelalarm Linie 1 <a href="#">Alarms_PaintShop[1]</a>	1	20.01.2014, 17:42:21, 359	20.01.2014, 17:42:40, 0	no
Sammelalarm Linie 2 <a href="#">Alarms_PaintShop[2]</a>	1	20.01.2014, 17:42:10, 859	no	no
Störung Sicherheitsystem Linie 1 <a href="#">Alarms_AssemblyShop[1]</a>	1	20.01.2014, 17:41:37, 609	no	no
Not Aus gedrückt <a href="#">Alarms_PaintShop[4]</a>	1	20.01.2014, 17:41:30, 609		

The bottom of the screen features a grey bar with two icons: a bar chart labeled 'Values' and a blue triangle with an exclamation mark labeled 'Alarms'.



## ALARMS

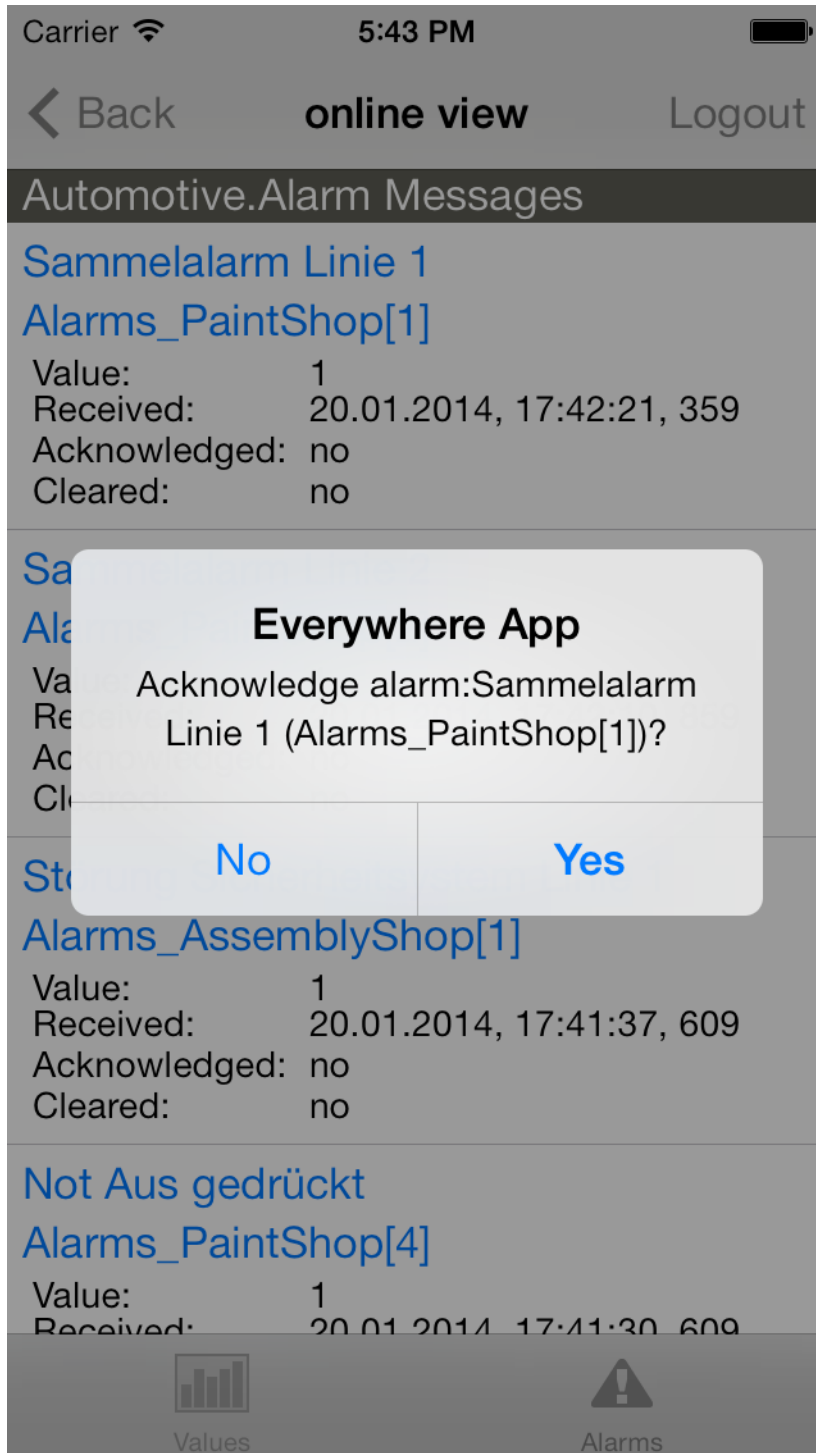
Parameters	Description
List with alarm entries	<p>List of active alarms:</p> <ul style="list-style-type: none"> <li>▶ Variable name</li> <li>Value</li> <li>Date and time of the alarm</li> <li>Acknowledged alarm: Yes/No</li> <li>Cleared alarm: Yes/No</li> </ul> <p>Tapping on an alarm entry acknowledges that entry.</p>

## FOOTER

Parameters	Description
<b>Values</b> (Symbol)	Switches to display of values of the variables. Blue background if active.
<b>Alarms</b> (Symbol)	Switches to display of alarms. Blue background if active.

### 7.4.9 Acknowledge alarms



An alarm can be acknowledged by tapping on the list entry.



**ACKNOWLEDGE AN ALARM WITH EVERYWHERE APP:**

1. Tap on the alarm entry in the alarm screen.
2. A dialog is opened.
3. Tap **yes** if you want to acknowledge the alarm.
4. **No** closes the acknowledgment dialog

After acknowledgment, the entry for the alarm (**Acknowledged:**) shows the date and time of acknowledgment.

Carrier 
5:44 PM


[< Back](#)
**online view**
[Logout](#)

Automotive.Alarm Messages

[Sammelalarm Linie 1](#)  
[Alarms\\_PaintShop\[1\]](#)  
Value: 1  
Received: 20.01.2014, 17:42:21, 359  
Acknowledged: 20.01.2014, 17:42:40, 0  
Cleared: no

---



[Sammelalarm Linie 2](#)  
[Alarms\\_PaintShop\[2\]](#)  
Value: 1  
Received: 20.01.2014, 17:42:10, 859  
Acknowledged: no  
Cleared: no

---

[Störung Sicherheitsystem Linie 1](#)  
[Alarms\\_AssemblyShop\[1\]](#)  
Value: 1  
Received: 20.01.2014, 17:41:37, 609  
Acknowledged: no  
Cleared: no

---

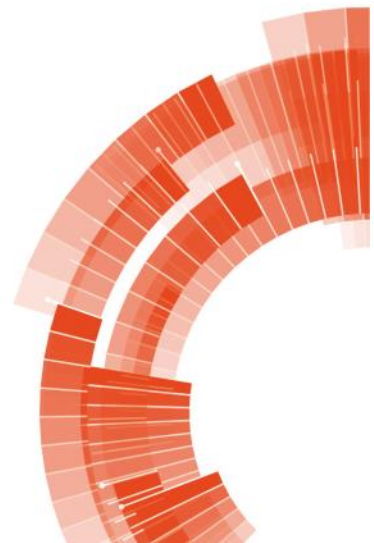
[Not Aus gedrückt](#)  
[Alarms\\_PaintShop\[4\]](#)  
Value: 1  
Received: 20.01.2014, 17:41:30, 609


Values

Alarms

## 7.5 Everywhere App for Desktop

### 7.5.1 Authentication

The login screen allows you to log in to an Everywhere Server by zenon and to select user profiles.



Parameters	Description
<b>Profile</b>	Name of the selected profile. <ul style="list-style-type: none"> <li>▶ Tapping or left-clicking opens the drop-down list to select an existing profile.</li> </ul>
<b>User name</b>	User name for login at the Everywhere Server.
<b>Password</b>	Password for login at the Everywhere Server.
<b>login</b> (Symbol)	Logs the app into Everywhere Server and switches to the next screen.

**Note:** Pay attention to capital letters and small letters when entering the data!


## APP BAR

Parameters	Description
<b>edit</b> (pencil symbol)	Tapping or left-clicking opens the screen to configure a profile.

## 7.5.2 Profile List

The profile overview lists existing profiles and administers these.

### ↶ Edit profile



## NAVIGATION

<b>&lt;-</b> (Symbol)	Switches to the previous screen.
--------------------------	----------------------------------

Parameters	Description
Profile List	<p>Lists existing profiles.</p> <ul style="list-style-type: none"> <li>▶ The profile that is currently selected is accentuated graphically.</li> <li>▶ Tapping or left-clicking on the desired profile selects the profile.</li> <li>▶ Tap or click the edit symbol to edit the selected profile.</li> </ul>

#### APP BAR

Parameters	Description
<b>Edit</b> (pencil symbol)	Switches the selected profile to editing mode.
<b>Add</b> (+ Symbol)	Creates a new profile.
<b>Delete</b> (Recycle Bin symbol)	Deletes the selected profile.
<b>Save</b> (Disk symbol)	Saves changes to the selected profile. <b>Note:</b> <i>Inactive</i> if no profile is being edited.
<b>Reset</b> (X-Symbol)	Discards all inputs. <b>Note:</b> <i>Inactive</i> if no profile is being edited.

#### TO EDIT AN EXISTING PROFILE:

- ▶ Tap or click on an existing profile from the list.
- ▶ Tap or click on the edit symbol in the app bar.

### 7.5.3 Edit Profile

In this screen, existing profiles can be changed and new profiles can be configured.

#### ← Edit profile



#### NAVIGATION

<-  
(Symbol)

Switches to the previous screen.



Parameters	Description
<b>Profile name</b>	Name of the profile: Freely selectable
<b>Server host name</b>	Name or IP address of the Everywhere Server to which a connection is to be created.
<b>Port HTTPS</b>	Port address for the connection to the Everywhere Server. Default: 8050
<b>Project name</b>	Project name for the connection to Everywhere Server.
<b>Display variables by:</b> (Drop-down list)	Display of the variables. Select from drop-down list: <ul style="list-style-type: none"> <li>▶ <b>Name</b> Variable display according to name</li> <li>▶ <b>Identification</b> Variable display according to identification</li> <li>▶ <b>Res. Label</b> Variable display according to resources label</li> </ul>
<b>Remember password:</b> (Slider)	Save password for input. The setting is also shown as text: <ul style="list-style-type: none"> <li>▶ <b>OFF:</b> (slider left and bright) Password for login is not saved in the profile settings</li> <li>▶ <b>On:</b> (slider right and dark) Password is saved in the profile settings</li> </ul> Default: OFF

## APP BAR

Parameters	Description
<b>Edit</b> (pencil symbol)	Switches the selected profile to editing mode.
<b>Add</b> (+ Symbol)	Creates a new profile.
<b>Delete</b> (Recycle Bin symbol)	Deletes the selected profile.
<b>Save</b> (Disk symbol)	Saves changes to the selected profile.
<b>Reset</b> (X-Symbol)	Discards all inputs.

## 7.5.4 Equipment model and equipment groups

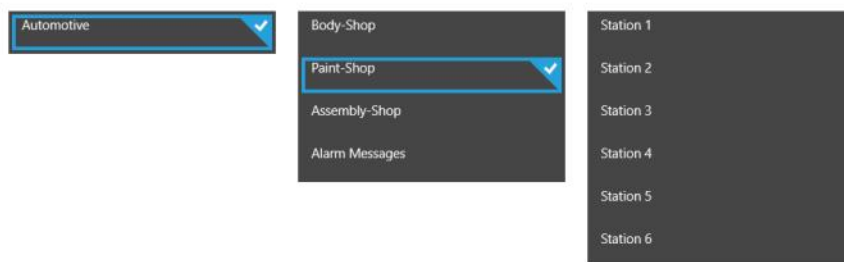
This screen shows the configured equipment model. Once an equipment model has been selected, the attendant equipment groups are displayed.

The attendant hierarchy is displayed by selecting an equipment group.

If a selected group contains variables, the view switches to the variable overview screen.

### ← variables groups

Automotive → Assembly-Shop → Paint-Shop



### APP BAR

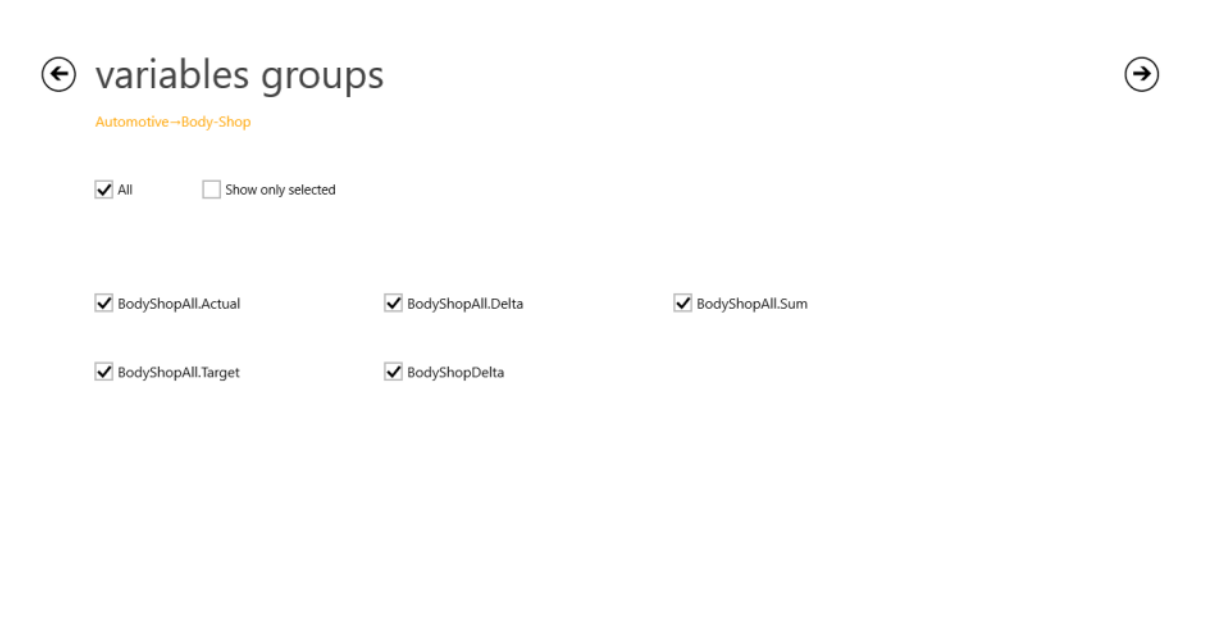
**Logout**  
(open lock symbol)

Ends connection to the server and returns to the login screen.



7.5.5 Variables selection

In this screen, configured variables from the selected equipment group are listed and selected.



Select the variables by tapping or left-clicking.

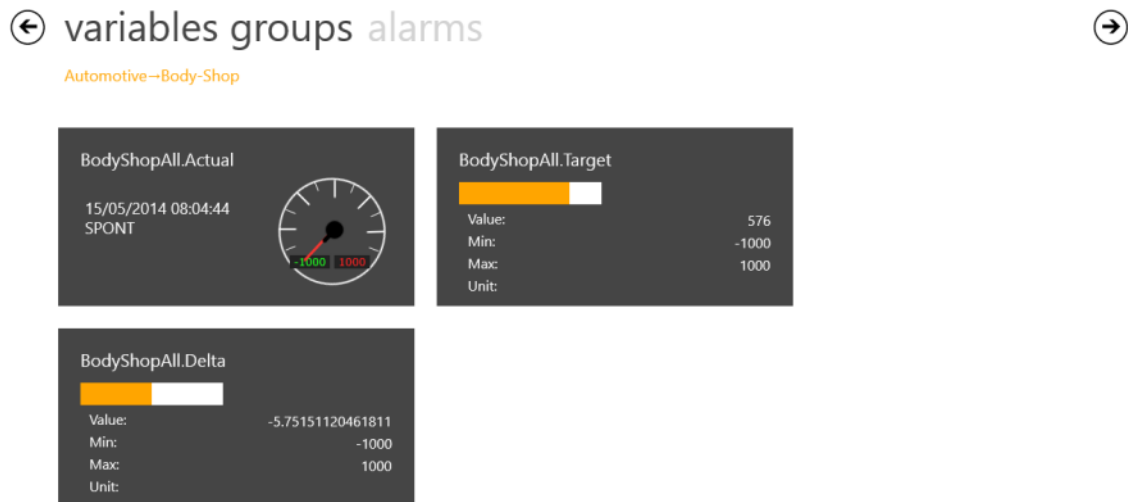
<b>&lt;-</b> (Symbol)	Switches to the previous screen.
<b>-&gt;</b> (Symbol)	Switches to the next screen.
<b>All</b>	Activates/deactivates all variables.
<b>Show only selected</b>	Only the selected variables are displayed in the screen of the variable view. Inactive if there is not at least one variables selected.

APP BAR

<b>Logout</b> (open lock symbol)	Ends connection to the server and returns to the login screen.
-------------------------------------	--

## 7.5.6 Values view

This screen visualizes the values of the selected variables. There are two views available.



Select the variables by left-clicking or tapping.

<b>&lt;-</b> (Symbol)	Switches to the previous screen.
<b>-&gt;</b> (Symbol)	Switches to the next screen.

#### VARIABLE LIST

Parameters	Description
Variable display	<p>Tapping or left-clicking on the variable switches between "normal view and "detailed view"</p> <ul style="list-style-type: none"> <li>▶ Normal view: <ul style="list-style-type: none"> <li>Variable name</li> <li>Variable value (numeric)</li> <li>Variable value (bar display)</li> <li>Min, max and unit</li> </ul> </li> <li>▶ Detail view: <ul style="list-style-type: none"> <li>Variable name</li> <li>Date and time of the last update</li> <li>Communication type</li> <li>Data type of the variable</li> <li>Value in numerical display</li> <li>Value with pointer instrument display</li> </ul> </li> </ul>

#### APP BAR

<b>Logout</b> (open lock symbol)	Ends connection to the server and returns to the login screen.
-------------------------------------	--

## 7.5.7 Alarm List

This screen visualizes active alarms in a list. Alarms can be acknowledged by tapping on the alarm entry.

### ⬅ variables groups alarms

Automotive--Alarm Messages

<b>Maintenance Screwdriver SC01</b> Alarms_AssemblyShop[5] Value: 1 Received: 15/05/2014 08:12:58 Cleared: No Acknowledged: No	<b>Alarm Screwdriver SC01</b> Alarms_AssemblyShop[3] Value: 1 Received: 15/05/2014 08:14:20 Cleared: No Acknowledged: No	<b>General Alarm Line 1</b> Alarms_PaintShop[1] Value: 1 Received: 15/05/2014 08:14:31 Cleared: No Acknowledged: No
<b>flow rate too low</b> Alarms_PaintShop[3] Value: 1 Received: 15/05/2014 08:14:06 Cleared: No Acknowledged: No	<b>Robot RS01 in Manual Mode</b> Alarms_BodyShop[3] Value: 1 Received: 15/05/2014 08:14:27 Cleared: No Acknowledged: No	<b>Emergency Stop Robot RS01</b> Alarms_BodyShop[1] Value: 1 Received: 15/05/2014 08:14:36 Cleared: No Acknowledged: No

Select the alarms by tapping or left-clicking.

<b>&lt;-</b> (Symbol)	Switches to the previous screen.
--------------------------	----------------------------------

## ALARMS

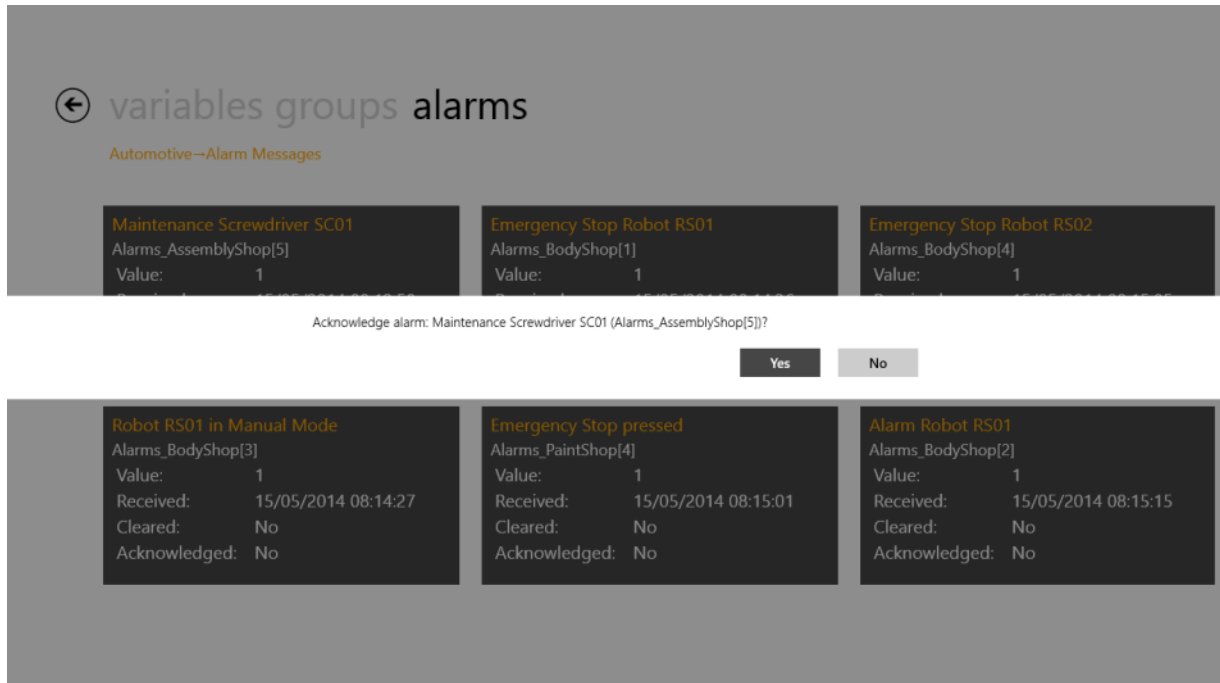
Parameters	Description
List with alarm entries	<p>List of active alarms:</p> <ul style="list-style-type: none"> <li>▶ Variable name</li> <li>Value</li> <li>Date and time of the alarm</li> <li>Acknowledged alarm: Yes/No</li> <li>Cleared alarm: Yes/No</li> </ul> <p>This alarm can be acknowledged by tapping or left-clicking.</p>

## APP BAR

<b>Logout</b> (open lock symbol)	Ends connection to the server and returns to the login screen.
-------------------------------------	--

### 7.5.8 Acknowledge alarms

An alarm can be acknowledged by tapping on the list entry.



#### ACKNOWLEDGING AN ALARM WITH EVERYWHERE APP FOR DESKTOP:

1. Tap or click on the alarm entry in the alarm screen.
2. A dialog is opened.
3. Tap or click **yes** if you want to acknowledge the alarm.
4. **no** closes the acknowledgment dialog

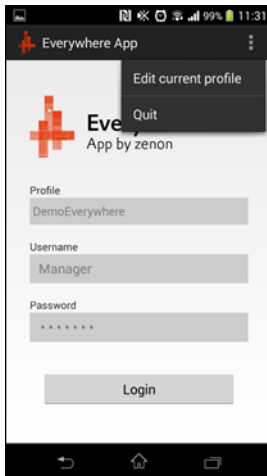
After acknowledgment, the entry for the alarm (**Acknowledged:**) shows the date and time of acknowledgment.



## 7.6 Everywhere App by zenon for Android

### 7.6.1 Authentication

The login screen allows you to log in to an Everywhere Server by zenon and to select user profiles.



#### ACTION BAR - ACTION BUTTONS

Parameters	Description
<b>Action Overflow</b> (symbol: three vertical dots)	Click the button to open a drop-down list of the actions.
<b>Edit current profile</b>	Opens the screen to configure a profile (on page 100).
<b>Quit</b>	Closes the app and switches back to the Start screen of the Android device.

Parameters	Description
<b>Profile</b>	Name of the selected profile.  Tapping on the profile name opens the dialog to select an existing profile.
<b>Username</b>	User name for login at the Everywhere Server.
<b>Password</b>	Password for login at the Everywhere Server.
<b>Login</b>	Logs the app into Everywhere Server and switches to the next screen.

**Note:** Pay attention to capital letters and small letters when entering the data!

#### APP BAR

Parameters	Description
<b>zurück</b> (symbol: curved arrow to the right)	Switches to the previous app screen.
<b>Home</b> (symbol: house)	Closes the app and switches back to the Start screen of the Android device.
<b>nn</b> (symbol: overlaid screens)	Overview of the last apps open on the device.

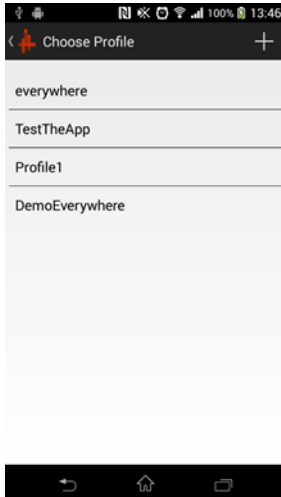


#### Information

The appearance and functionality of the app bar is the same on all screens. This bar is very often already integrated into the hardware of an Android device. In this case, app bar is displayed.

## 7.6.2 Profile List

The profile overview lists existing profiles and administers these.



### ACTION BAR - ACTION BUTTONS

Parameters	Description
< (Symbol)	Switches back one screen.
+ (Symbol)	Creates a new profile and switches to the <b>Edit profile</b> screen.

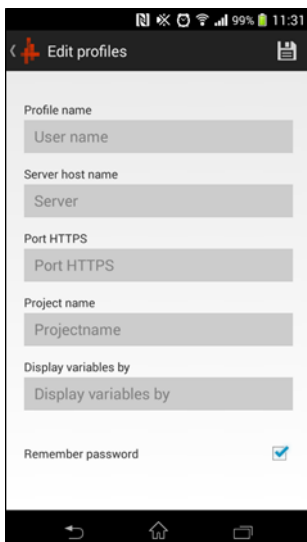
Parameters	Description
Profile List	<p>Lists existing profiles.</p> <p>A long click on an entry opens a context menu:</p> <ul style="list-style-type: none"> <li>▶ <code>Edit profile</code> Switches to the profile editing screen. Allows changes to an existing profile.</li> <li>▶ <code>Delete profile</code> Deletes selected profile.</li> </ul> <p>Tapping on the desired profile activates the profile and switches to the login screen.</p>

#### TO EDIT AN EXISTING PROFILE:

- ▶ Tap an existing profile from the list.  
**Attention:** Ensure that you tap on the list entry for a long time.
- ▶ Select the **Edit profile** entry in the context menu that appears.
- ▶ You switch to the **Edit profile** screen.

### 7.6.3 Edit Profile

In this screen, existing profiles can be changed and new profiles can be configured.



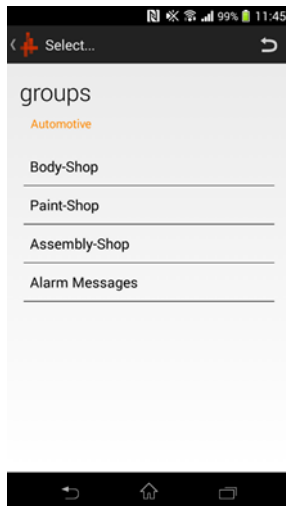
**ACTION BAR - ACTION BUTTONS**

Parameters	Description
< (Symbol)	Switches back one screen.
<b>Save</b> (Disk symbol)	Saves the current settings and returns to <b>editing the profiles</b> .

Parameters	Description
<b>Profile name</b>	Name of the profile: Freely selectable
<b>Server host name</b>	Name of the Everywhere Server to which a connection is to be created.
<b>Port HTTPS</b>	Port address for the connection to the Everywhere Server.  Default: 8050
<b>Project name</b>	Project name for the connection to Everywhere Server.
<b>Display variables by:</b>	Display of the variables. Select from drop-down list: <ul style="list-style-type: none"> <li>▶ <b>Name</b> Variable display according to name</li> <li>▶ <b>Identification</b> Variable display according to identification</li> <li>▶ <b>Res. Label</b> Variable display according to resources label</li> </ul>
<b>Remember password:</b> (Checkbox)	Save password for input. The setting is also shown as text: <ul style="list-style-type: none"> <li>▶ <b>inactive:</b> Password for login is not saved in the profile settings</li> <li>▶ <b>active:</b> Password is saved in the profile settings.</li> </ul> Default: inactive.

## 7.6.4 Equipment model and equipment groups

This screen shows the configured equipment model. Tapping on an equipment model shows the equipment groups contained therein. Tapping on an equipment group name switches to the next screen with the selection of the allocated variables.



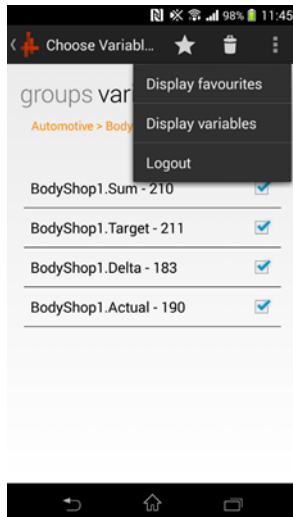
### ACTION BAR - ACTION BUTTONS

Parameters	Description
< (Symbol)	Switches back one screen.
<b>Logout:</b> (symbol: curved arrow to the left)	Ends connection to the server and returns to the login screen.

Parameters	Description
[name of equipment group]	For easier orientation, either the equipment model name or the equipment group name - depending on the selection - is displayed in orange text above the list.
List of the configured equipment groups or variables.	Lists existing profiles:  Clicking on the equipment group name switches to the overview of configured equipment groups.  Clicking on an equipment group name switches to the variable overview and lists all variables configured in the equipment group in a list.

### 7.6.5 Variables selection

In this screen, configured variables from the selected equipment group are listed and selected.



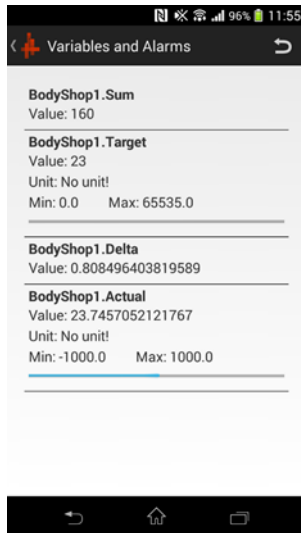
**ACTION BAR - ACTION BUTTONS**

Parameters	Description
< (Symbol)	Switches back one screen.
Favorites (star symbol)	Adds the selected variable(s) to the list of favorites.
Delete favorites (Recycle Bin symbol)	Deletes all variables from the list of favorites.
<b>Action Overflow</b> (symbol: three vertical dots)	Click the button to open a drop-down list of the actions.
<b>Display favorites</b>	Displays all variables from the list of favorites in a list.
<b>Display variables</b>	Displays all activated variables of the selected equipment group in a list. To activate a variable, the checkbox in the right column must be activated.
<b>Logout</b>	Ends connection to the server and returns to the login screen.
Display all	Selects all listed variables.  This is visualized by a blue tick in the checkbox next to the variable name.



### 7.6.6 Values view

This screen visualizes the values of the selected variables. There are two views available.



#### ACTION BAR - ACTION BUTTONS

Parameters	Description
< (Symbol)	Switches back one screen.
<b>Logout:</b> (symbol: curved arrow to the left)	Ends connection to the server and returns to the login screen.

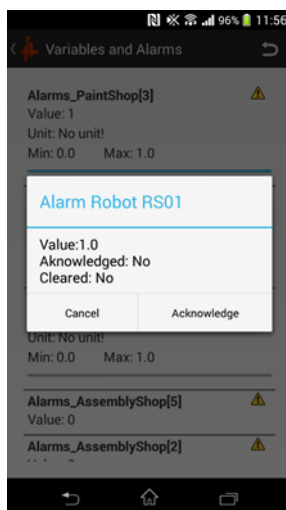
## VARIABLE LIST

Parameters	Description
Variable list	<p>List of selected variables:</p> <p>Clicking on the variable switches between "normal view" and "detailed view"</p> <ul style="list-style-type: none"> <li>▶ Normal view: <ul style="list-style-type: none"> <li>Variable name</li> <li>Variable value (numerical)</li> </ul> </li> <li>▶ Detailed view: <ul style="list-style-type: none"> <li>Variable name</li> <li>Variable value</li> <li>Minimum and Maximum value</li> <li>Value in bar graph display</li> </ul> </li> </ul>

### 7.6.7 Acknowledging alarm list and alarms

The alarms are integrated directly into the value view of the variables. If there is an alarm, a warning sign is shown next to the variable in the value view screen.

A long click on the variable entry opens the attendant alarm entry. The alarm can be confirmed in the subsequent dialog.



**ACTION BAR - ACTION BUTTONS**

Parameters	Description
< (Symbol)	Switches back one screen.
<b>Logout:</b> (symbol: curved arrow to the left)	Ends connection to the server and returns to the login screen.

**ALARMS**

Parameters	Description
List with alarm entries	<p>The word Alarms_ is placed in front of the variable name.</p> <p>A long tap on an alarm entry opens the dialog to acknowledge an alarm.</p>

**CONFIRM ALARM DIALOG**

Parameters	Description
<b>Variablenname</b>	Name of the variable with pending alarm. The word "Alarm_" is placed in front of the variable name.
<b>Wert</b>	Value of the variable.
<b>Acknowledged:</b>	<p>Alarm confirmed:</p> <ul style="list-style-type: none"> <li>▶ No : Alarm has not yet been confirmed</li> <li>▶ Yes : Alarm has already been confirmed</li> </ul> <p>Default: No</p> <p>Note: This field cannot be changed.</p>
<b>Clear</b>	<p>Alarm ended:</p> <ul style="list-style-type: none"> <li>▶ No : The alarm state still keeps occurring.</li> <li>▶ Yes : The alarm state no longer occurs.</li> </ul> <p>Default: No</p> <p>Note: This field cannot be changed.</p>

Cancel	Closes the dialog and returns to the value view of the variables.
Acknowledge	Confirms alarm and closes the dialog.

#### ACKNOWLEDGING AN ALARM WITH EVERYWHERE APP:

1. Click on the alarm entry in the **value view** for a long time.
2. A dialog is opened.
3. Tap **Acknowledge** if you want to acknowledge the alarm.
4. **Cancel** closes the acknowledgment dialog

After acknowledgment, the entry for the alarm (**Acknowledged:**) shows the date and time of acknowledgment.

## 8. Notifier App by zenon

The **Notifier App by zenon** makes it possible to acknowledge alarms that are sent by means of **Message Control** as an SMS to a **Smartphone**.

Note: The **App** is only available for devices with the **Android** operating system.



### License information

*The **Notifier App by zenon** is freely available. It works together with the **Message Control** module. This requires a license.*

### 8.1 Functionality

The **Notifier App by zenon** displays SMSs that have been sent by the **Message Control** module and contain certain key words or have been sent by certain telephone numbers. After receipt, an alarm is played back on the **Smartphone** for 30 seconds.

If the message is opened within the time period, the user can acknowledge the alarm by SMS or reject it. If the message is ignored, then the information is filed to the **Notification-Bar**. If this is activated, the acknowledgment screen is called up again.

## 8.2 Configuration

Key words and telephone numbers, as well as PIN and NA can be set up when the App is started.

To use the App:

1. Download the **Notifier App by zenon** from the store.
2. On the **Homescreen**, open the **Notifier App**.



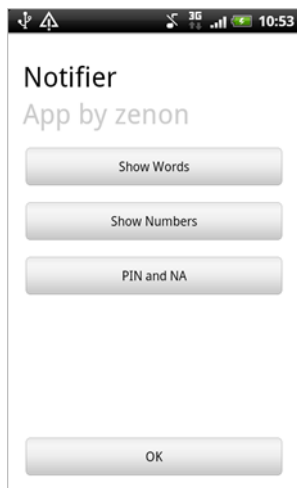
3. The configuration window is shown.

### CONFIGURE APP

The configuration screen allows the configuration of:

- ▶ Keywords
- ▶ Telephone numbers from which SMSs are received

► PNA and NA

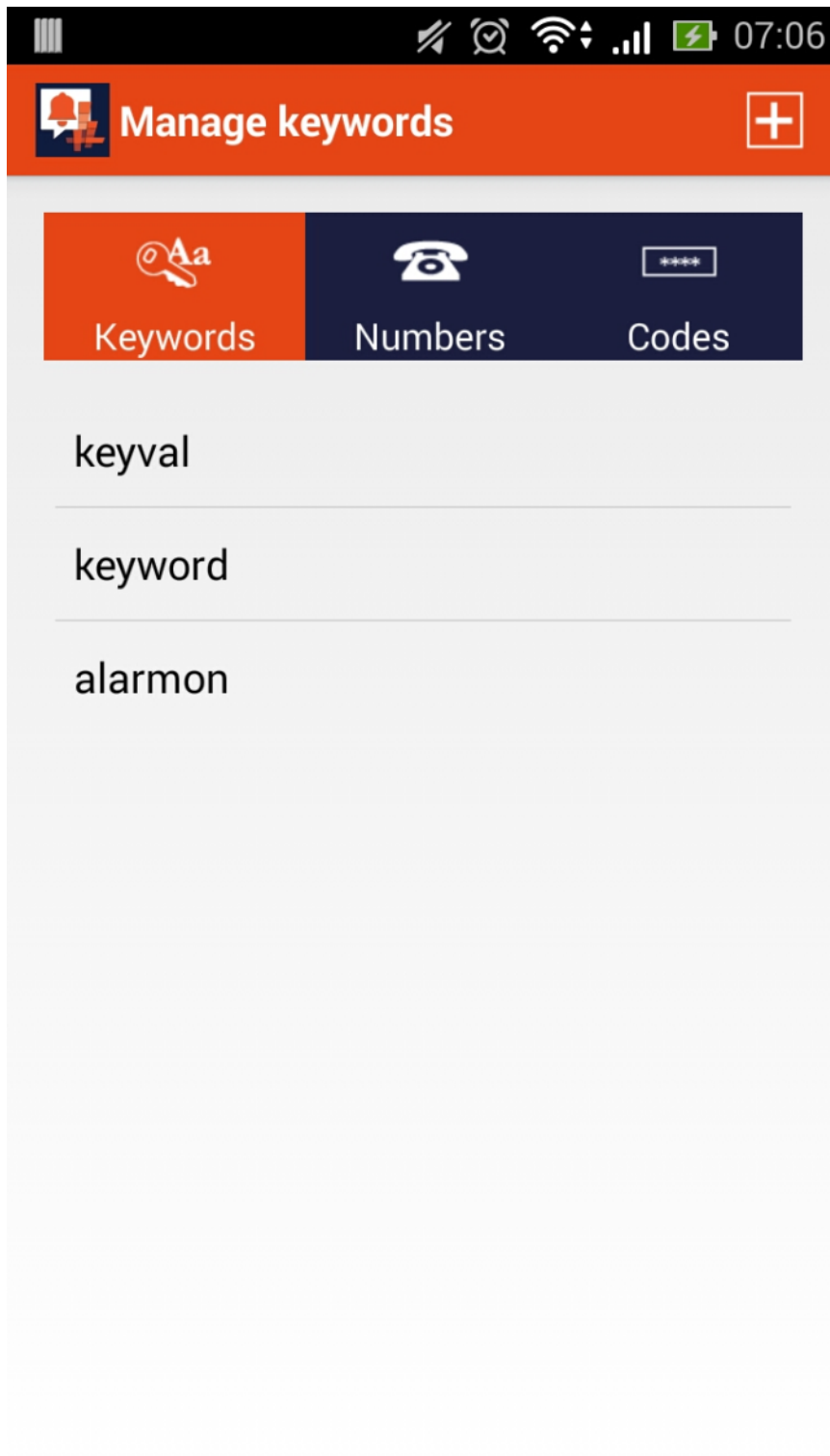


### CONFIGURE KEYWORDS

To configure keywords:

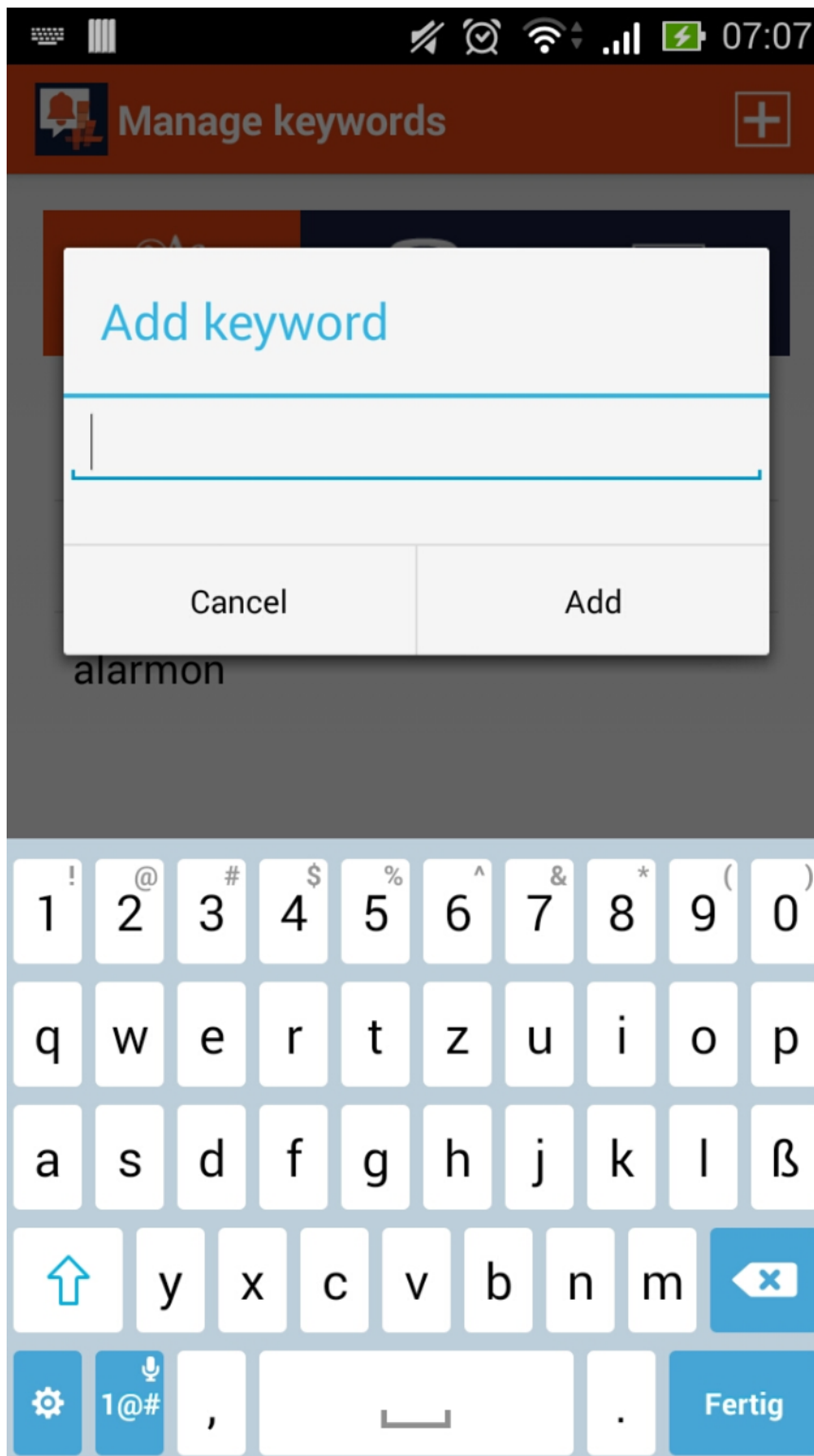
1. Click on the button **Show Words**.

2. A list of configured key words is shown.



3. If a key word is selected, it can be deleted by means of a dialog.

4. To accept a new key word, click on the **Add New Item** button.
5. The dialog to add it is opened.





6. Enter the key word and click on **Add**.

### **CONFIGURE TELEPHONE NUMBERS**

To configure telephone numbers from which SMSs can be displayed:

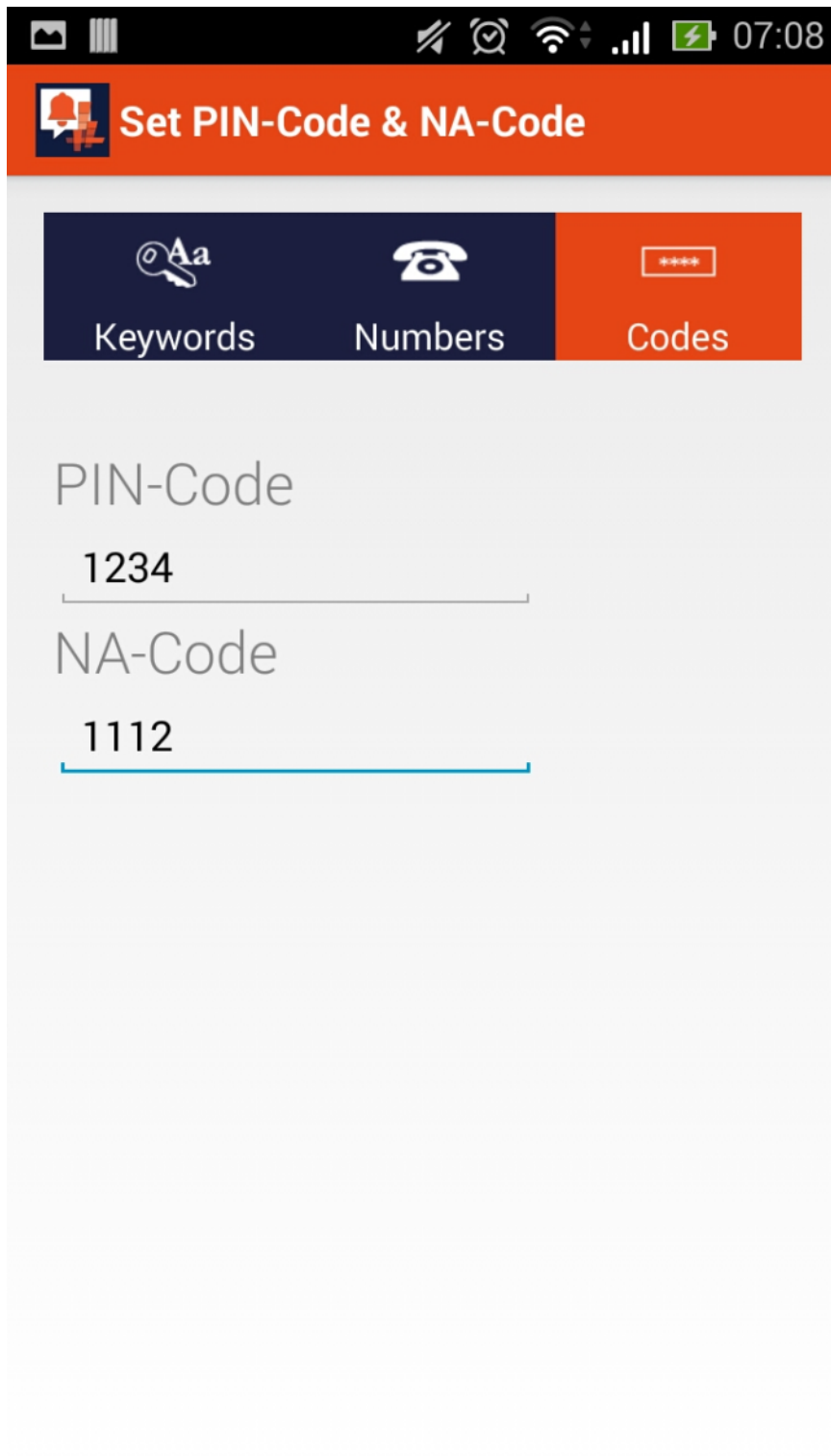
1. Click on the button **Show Numbers**.
2. A list of the configured telephone numbers is displayed.
3. If a telephone number is selected, it can be deleted by means of a dialog.
4. To accept a new telephone number, click on the **Add New Item** button.
5. The dialog to add it is opened.
6. Enter the telephone number and click on **Add**.

### **CONFIGURE PIN AND NA**

To configure PIN and NA:

1. Click on the **PIN and NA** button.

2. The dialog to enter the PIN and NA is opened.



3. Enter the necessary PIN for acknowledgment.

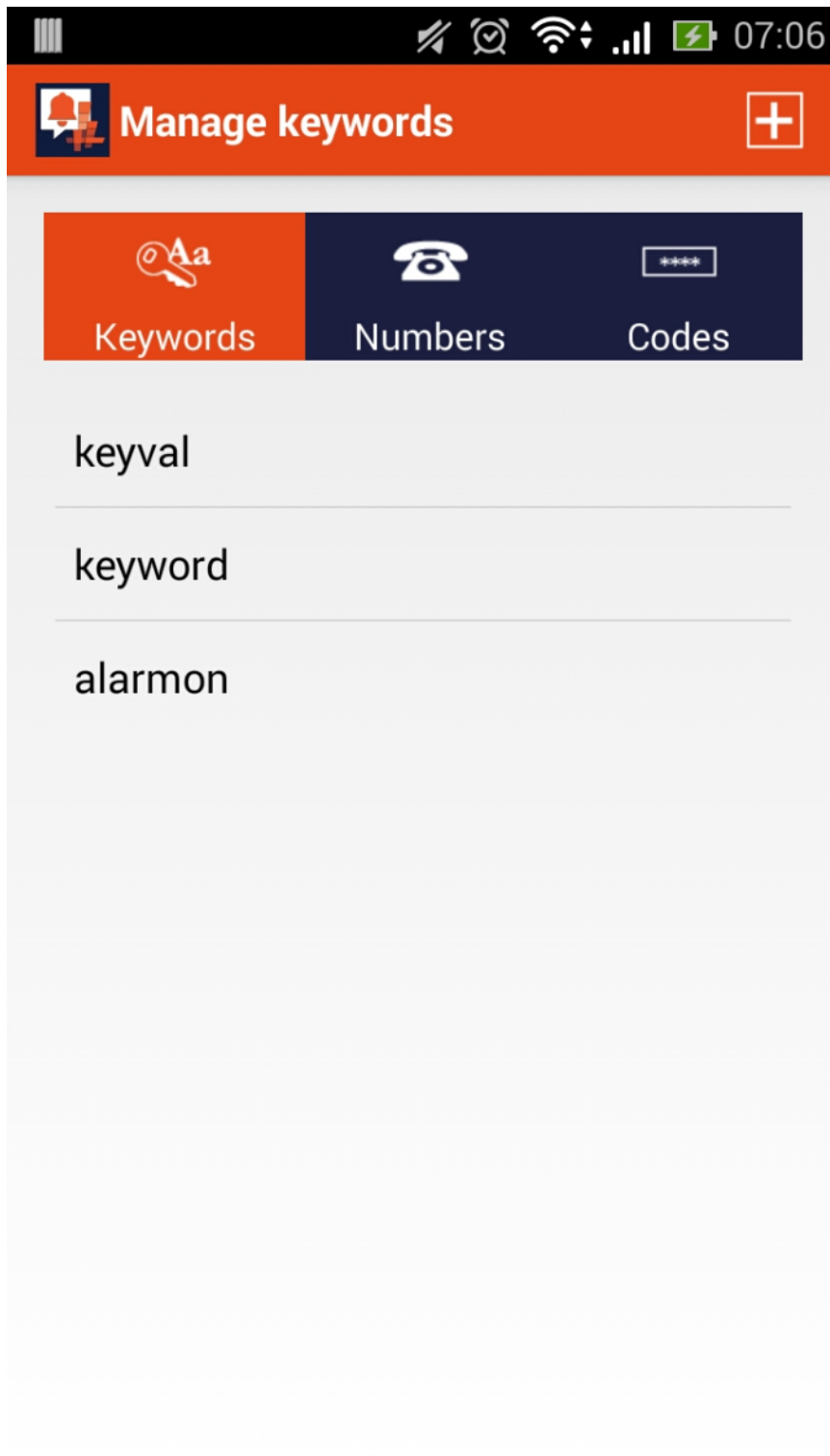
4. Enter the necessary NA for rejecting the alarm.
5. Click on **OK**.

### 8.2.1 Configure keywords

To configure keywords:

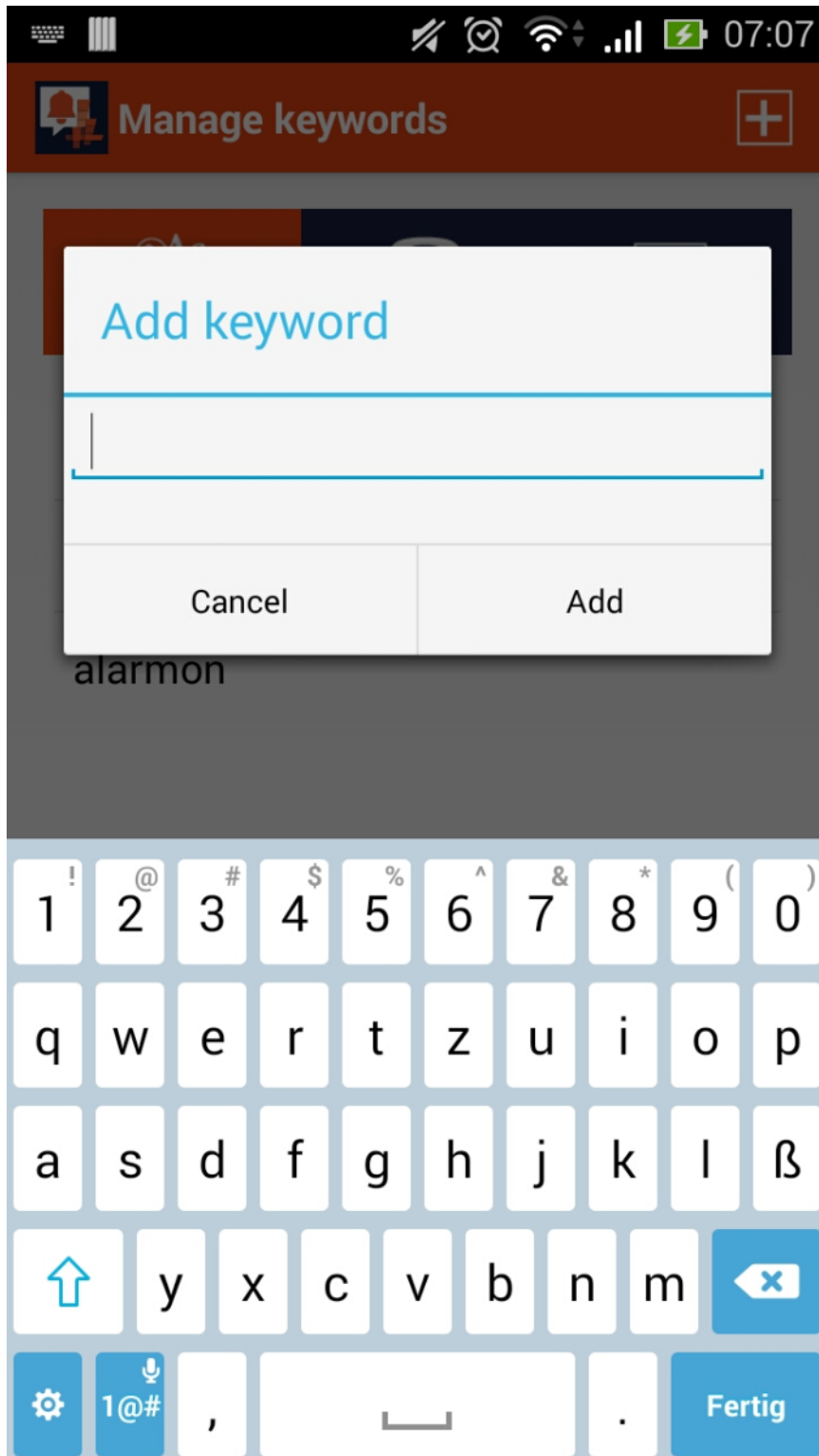
1. Click on the button **Show Words**.

2. A list of configured key words is shown.



3. If a key word is selected, it can be deleted by means of a dialog.

4. To create a new key word, click on the + button.  
Note: With **Android** version <3.0, adding is carried out by clicking on the **Add New Item** button.
5. The dialog to add it is opened.



6. Enter the key word and click on **Add**.

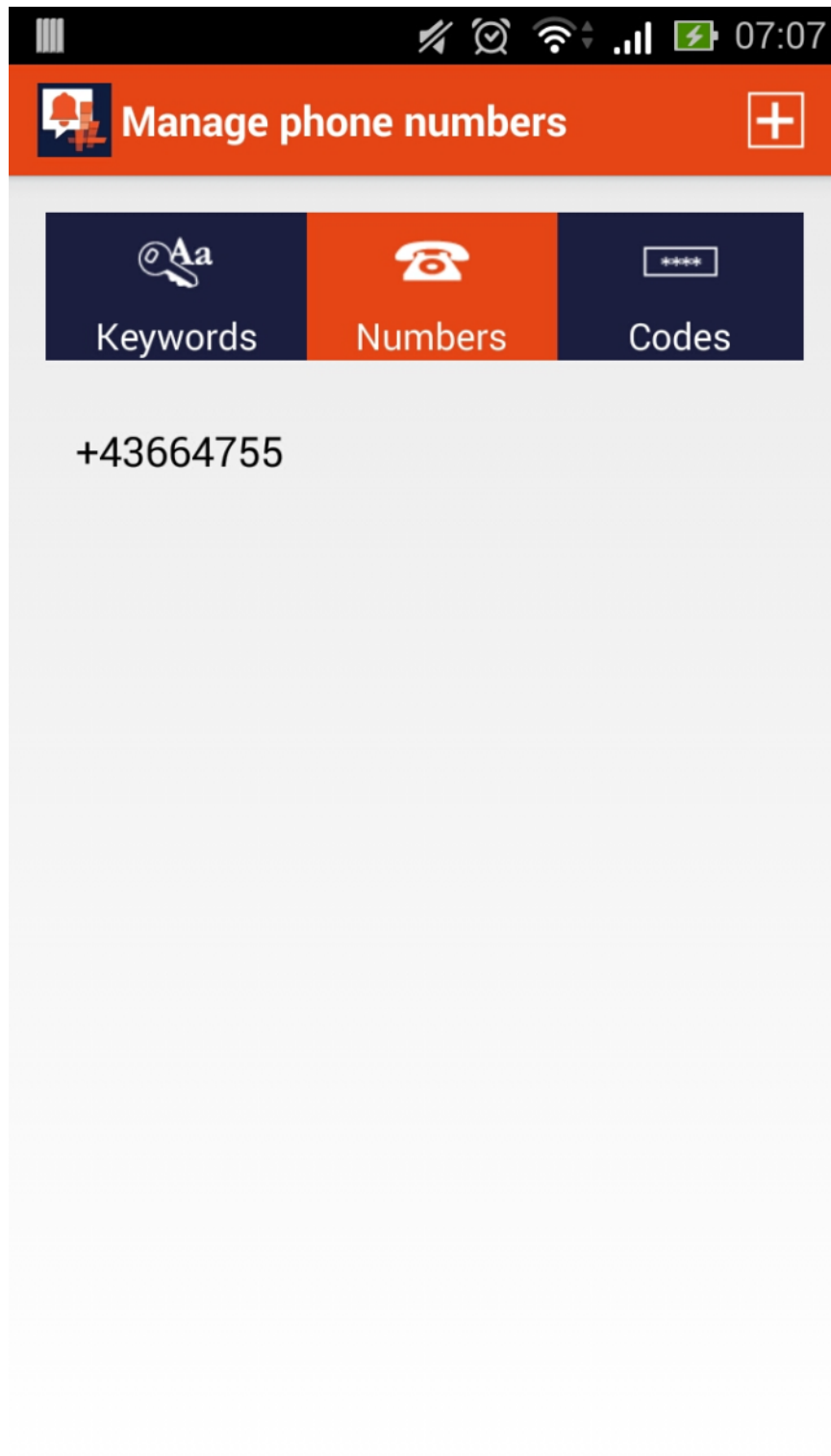
**Note:** Only one single keyword can be entered. Spaces are not permitted:

### 8.2.2 Configure telephone numbers

To configure telephone numbers from which SMSs are displayed:

1. Click on the button **Show Numbers**.

2. A list of the configured telephone numbers is displayed.



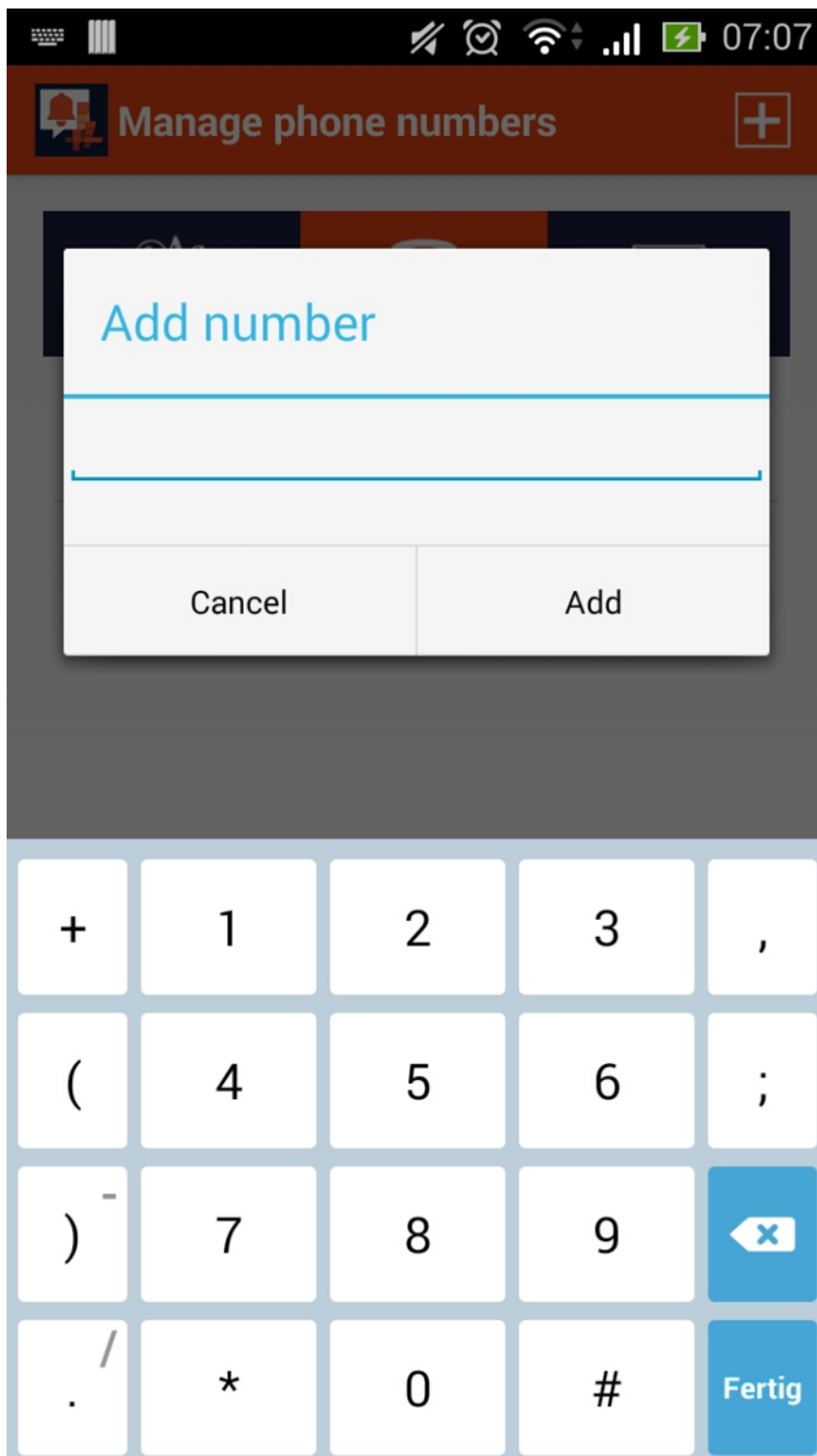
3. If a telephone number is selected, it can be deleted by means of a dialog.

4. To create a new telephone number, click on the + button.

Note: With **Android** version <3.0, adding is carried out by clicking on the **Add New Item** button.



5. The dialog to add it is opened.



The screenshot shows a mobile application interface. At the top, a status bar displays various icons and the time 07:07. Below it, a dark red header bar contains the text 'Manage phone numbers' and a plus icon. The main content area is grey. A white dialog box titled 'Add number' is centered on the screen. It has a text input field and two buttons at the bottom: 'Cancel' and 'Add'. Below the dialog box, a numeric keypad is visible. The keypad has a grid of buttons: a plus sign, digits 1-9, a comma, a left parenthesis, digits 4-6, a semicolon, a right parenthesis with a minus sign, digits 7-9, a blue back arrow button with an 'x', a period/slash, an asterisk, the digit 0, a hash sign, and a blue button labeled 'Fertig'.

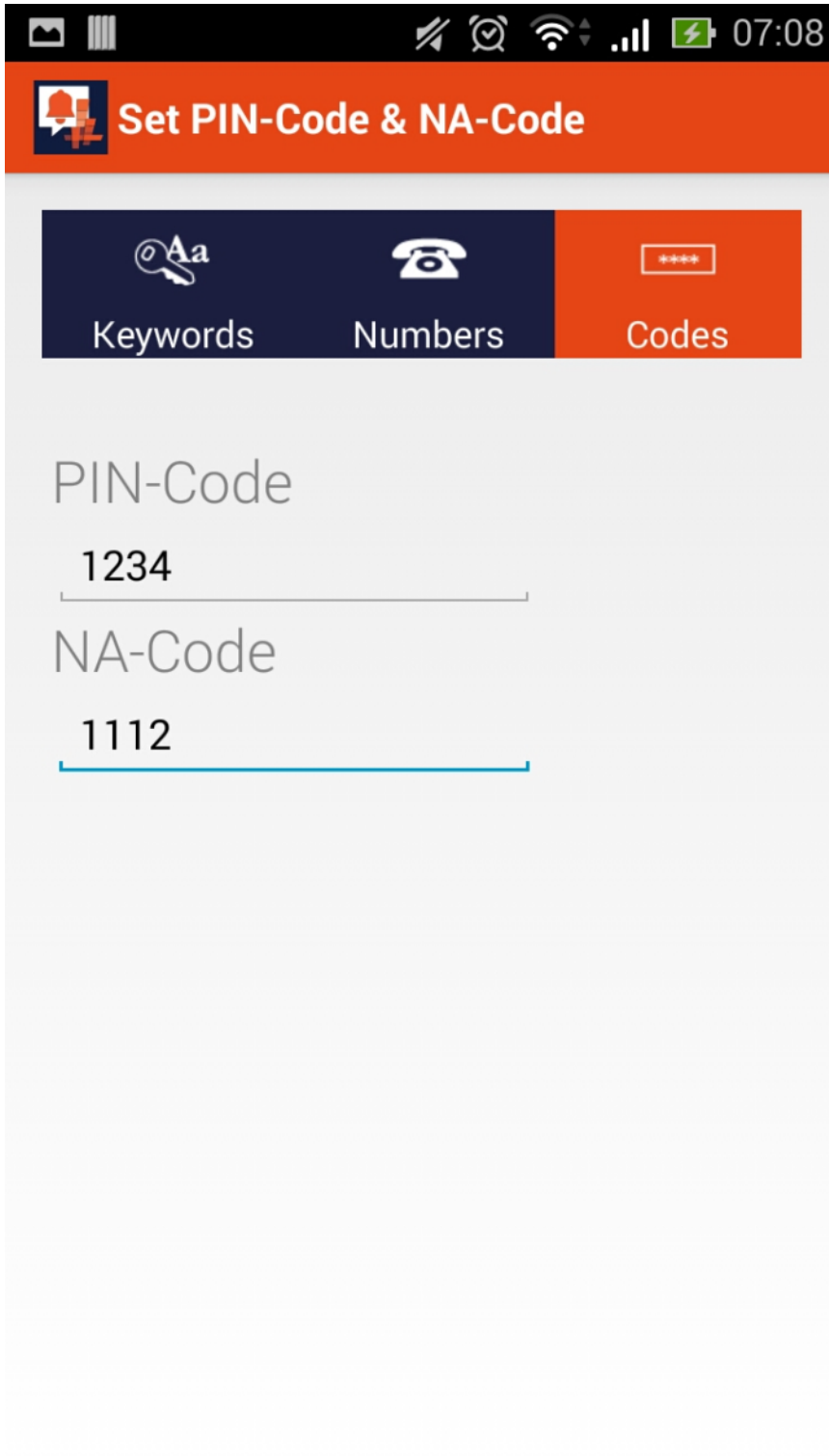
6. Enter the telephone number and click on **Add**.

Attention: Telephone numbers must be entered in the format +[country code] [number].  
Example: +4369912345678

### 8.2.3 Configure PIN and NA code

To configure PIN and NA code:

1. Click on the button `codes`.



Set PIN-Code & NA-Code

Keywords Numbers Codes

PIN-Code

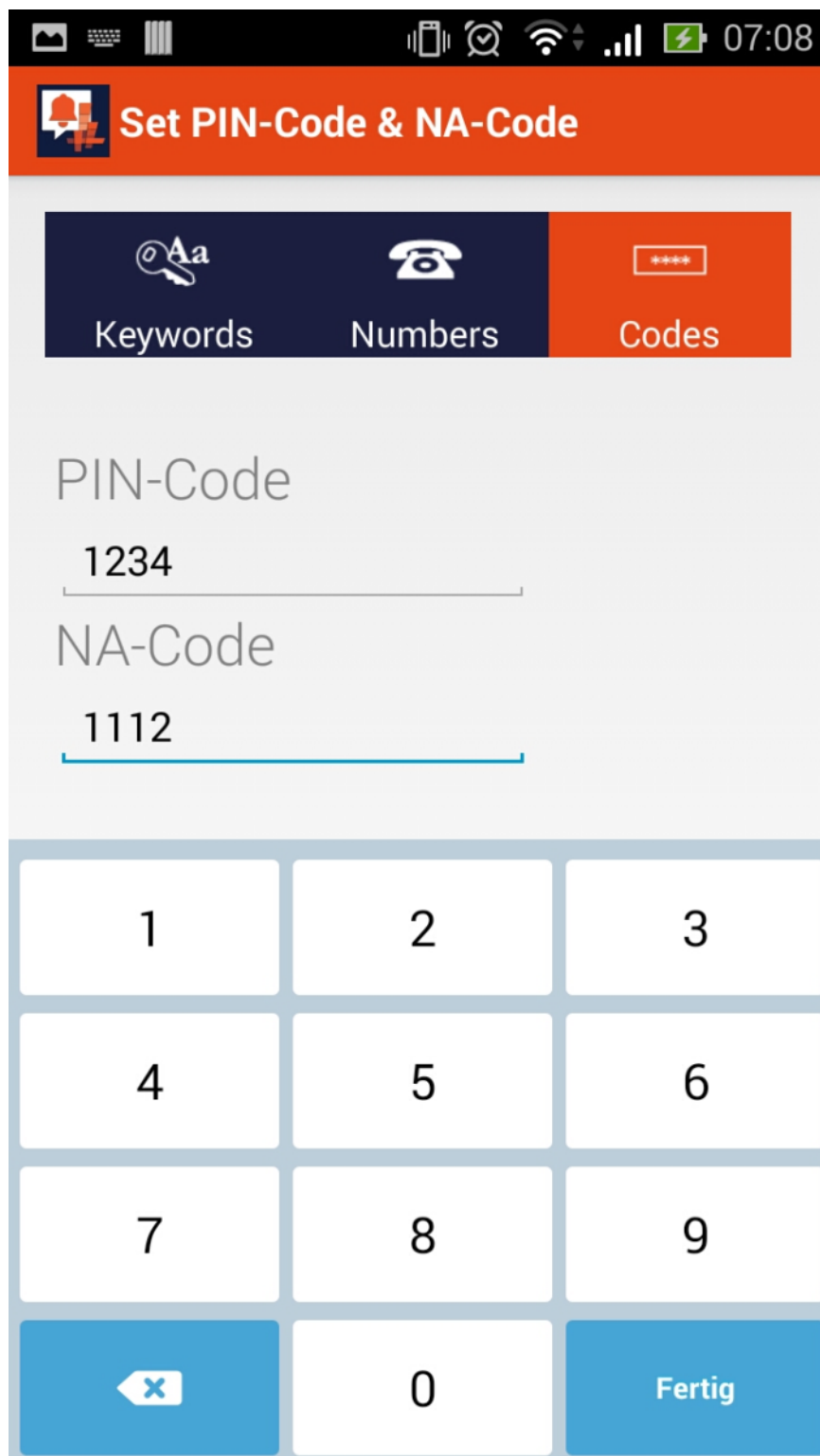
1234

NA-Code

1112

2. The codes for PIN and NA are shown.

3. To enter or change a code, click in the input field.
4. The touch keyboard to enter the PIN and NA is opened.



Set PIN-Code & NA-Code

Keywords Numbers Codes

PIN-Code  
1234

NA-Code  
1112

1 2 3  
4 5 6  
7 8 9  
⬅ 0 Fertig

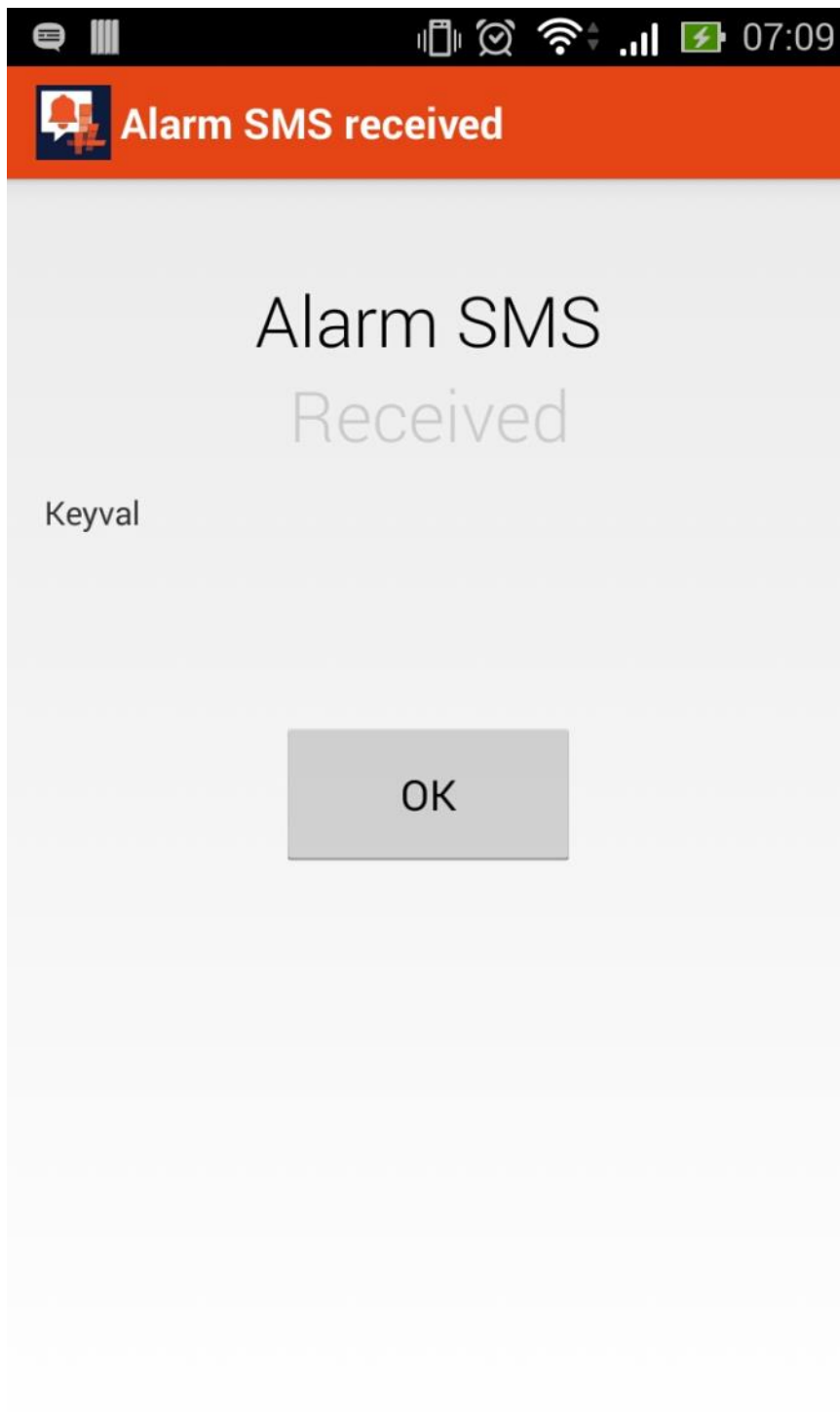
5. Enter the 4-digit PIN required for acknowledgment.
6. Enter the 4-digit NA required for rejecting the alarm.
7. Click on OK.

## 8.3 Receive and acknowledge alarm SMS

### RECEIVE ALARM SMS

Procedure for receipt of SMS:

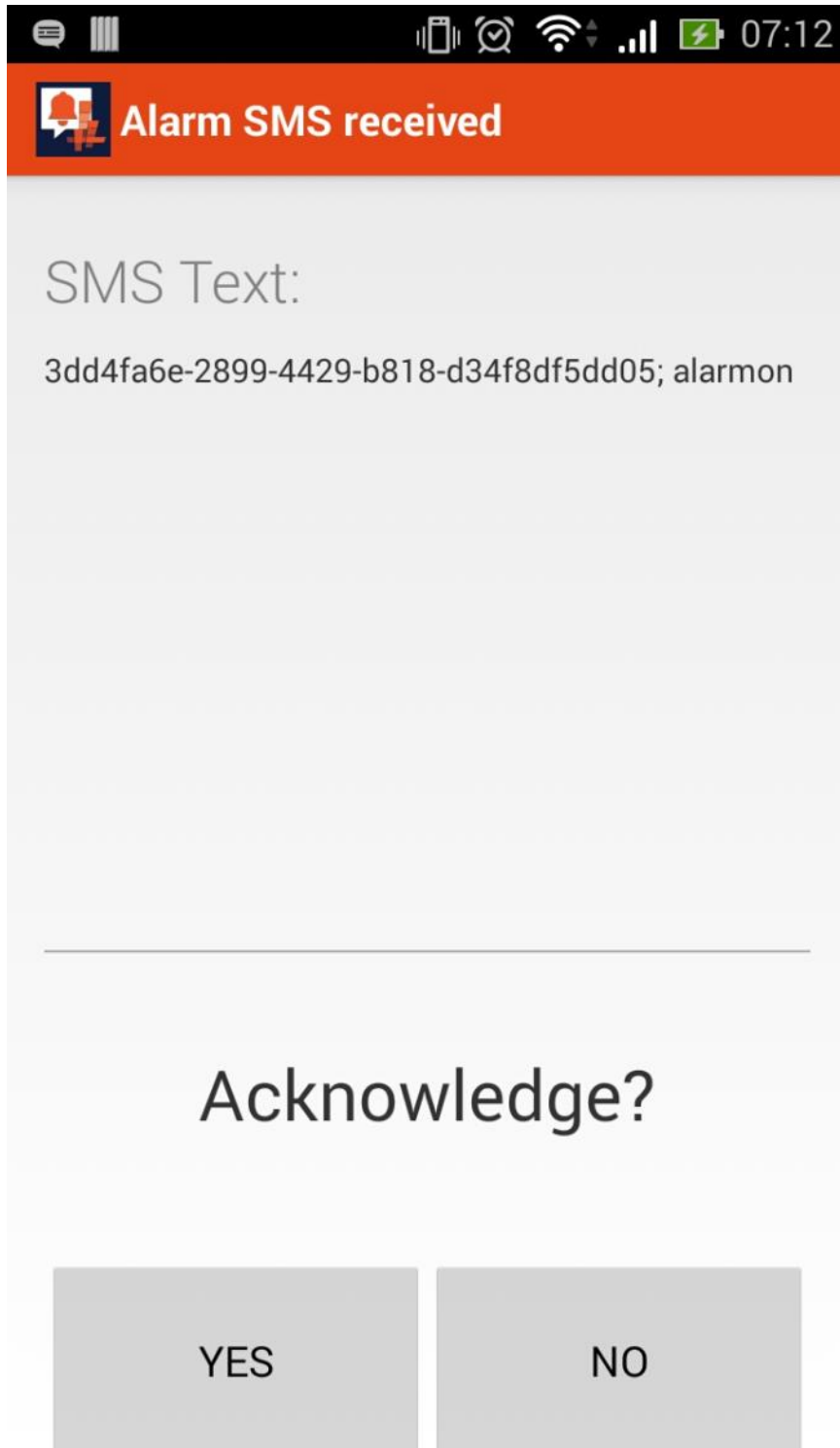
1. The **Notifier App by zenon** checks all incoming SMSs for key words and sender telephone number.
2. If the telephone number or a word in the text of the SMS correspond to the set parameters, an alarm is played back for 30 seconds and a notice window is opened.



3. To accept the alarm SMS and to react to this, click on **OK**.
4. If you do not react within 30 seconds, the window is closed, the alarm is ended and information is filed to the **Notification-Bar**.

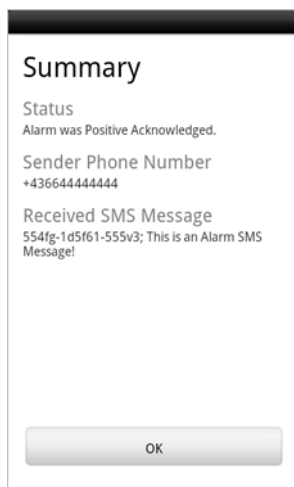
## RESPOND TO ALARM SMS

If the alarm SMS is received, a dialog with the message and response possibilities is opened.



Parameters	Description
<b>Positive Acknowledge</b>	Acknowledges the alarm. An SMS in <b>Alarm-ID;PIN</b> format is sent. The PIN must be stored in the configuration to do this.
<b>Negative Acknowledge</b>	Declines acknowledgment of the alarm. An SMS in <b>Alarm-ID;NA</b> format is sent. The NA must be stored in the configuration to do this.
<b>SMS Text:</b>	Display of the text of the SMS.
<b>Later</b>	Clicking on the button opens the dialog without a reaction. Information is stored in the <b>Notification-Bar</b> .

After an action has been executed, a window with the summary of the last-selected action is displayed.



Close the App by clicking on OK.



### Information

You can also find details on acknowledging the **Message Control** module in the **Acknowledging messages** chapter in the **Message Control** manual.

## EXECUTE THE APP FROM THE NOTIFICATION BAR ONCE AGAIN

If the alarm is ignored or the **Notifier App by zenon** is ended without an action, information is sent to the **Notification-Bar**.



To react to this message:

1. Click on the alarm symbol.



2. Select the alarm entry.



3. The **Notifier App by zenon** is opened with the window to answer an SMS.