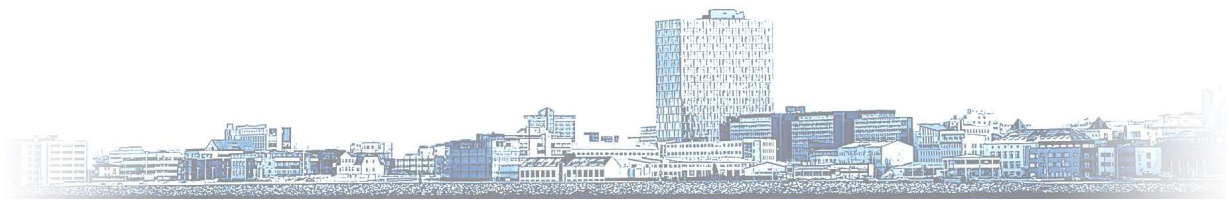
LS Retail
Mobile Inventory 1.1

Install and Quick Guide



© Copyright 2014, LS Retail ehf. All rights reserved. All trademarks belong to their respective holders

Contents

1 Introduction 1

2 System Overview 2

2.1 System requirements 2

3 Installation of WCF web service 2

3.1 Create SQL Server permissions in the NAV database 2

3.2 INVService – a WCF web service running under IIS 3

4 Installation of the Mobile Inventory application 5

4.1 Installation on mobile device 5

4.2 Activation 7

5 Functionality 9

5.1 Startup Menu 9

5.2 Utility 10

5.3 Lookup 11

5.3.1 Scan or local search 11

5.4 Worksheets 13

5.4.1 Stock Management 13

5.4.2 Negative Adjustment 13

5.5 Documents 15

5.5.1 Receive Menu 15

5.5.2 Picking Menu 16

6 NAV configuration for mobile inventory 18

6.1 General setup for Mobile in NAV 19

6.2 Terminal 25

6.3 Staff 27

6.4 Quick Scan 28

Appendix A 30

Appendix B 35

Appendix C 38

Appendix E 40

Installing IIS and WCF 40

Appendix F 43

LS Omni Mobile INVService Installer package 43

LS Retail ehf.
Katrinartun 2, 105 Reykjavik, Iceland
Tel: +354 414 5700 Fax: +354 571 2728

Revision History

No.	Date	Changes
1.0	2012	First version
2.0	March 2013	Same functionality except now connects to NAV 7
3.0	May 2013	Disclaimer in System Requirements chapter
4.0	June 2013	Minor corrections in text
5.0	October 2013	Updated text for SP6
6.0	January 2014	New chapter Quick Scan
7.0	February 2014	Updated for version 1.1

LS Retail ehf.
Katrinartun 2, 105 Reykjavik, Iceland
Tel: +354 414 5700 Fax: +354 571 2728

1 Introduction

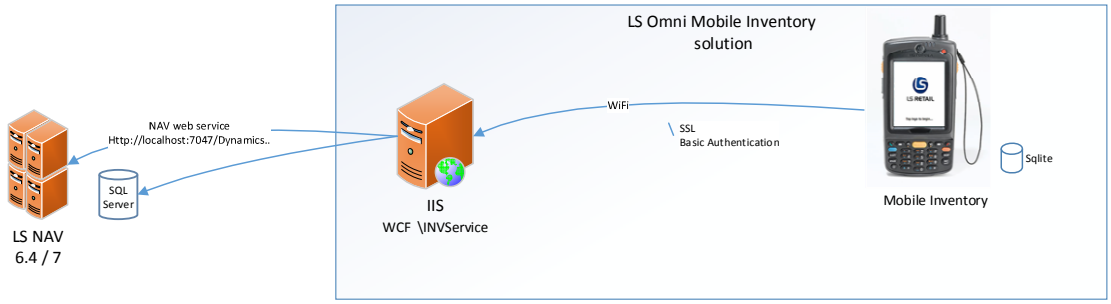
The purpose of this document is to give a quick overview of LS Retail's mobile solution, the LS Mobile Inventory solution, as part of the LS NAV 6.4 and 7.0 release.

LS Mobile Inventory is a highly adaptable solution where the functionality of the mobile application is controlled by NAV. Once the mobile application has been installed on the device it uses the NAV web service to retrieve and store its configuration locally. Hence NAV must be correctly configured for the mobile application to function properly.

The first part of this document will focus on the mobile application, installation and functionality. The second part will focus on the LS NAV Inventory Management.

2 System Overview

The LS Omni Mobile Inventory solution consists of the Mobile Inventory app that runs on Windows Mobile device and a WCF web service (called INVService) running under IIS 7 and above. The INVService calls LS NAV web services and handles the replication between NAV SQL Server and the local datastore on the mobile device.



2.1 System requirements

The LS Omni Mobile Inventory solution requires LS Retail NAV 6.4. The LS Omni Mobile Inventory WCF web server is fully supported and tested on the following operating systems.

- Windows 7 / Windows 8 / Windows Server 2008 / 2012
- IIS 7.0 or above must be installed
- Microsoft .Net 4.0 framework must be installed

The LS Omni Mobile Inventory application client runs on Windows Mobile 6.1 (and above), Windows CE 6.0 (and above) and requires .Net Compact Framework 3.5.

The installation requires 4 files that are part of this release.

- LSRetailMobileInventory.CAB
  - Installs the Mobile Inventory app on the mobile device
- LSOmni.Inventory.NAVDBSetup.exe
  - Creates a SQL Server user (INVUser) and grant permission on NAV tables
- LSOmni.Inventory.WcfServiceSetup.msi
  - Creates a WCF web service called INVService under IIS 7
- MobileInventoryGenerator.exe
  - Used to generate a password used by INVService to connect to NAV

3 Installation of WCF web service

The LS Omni Mobile Inventory solution requires permission in the NAV SQL Server database and a WCF web service

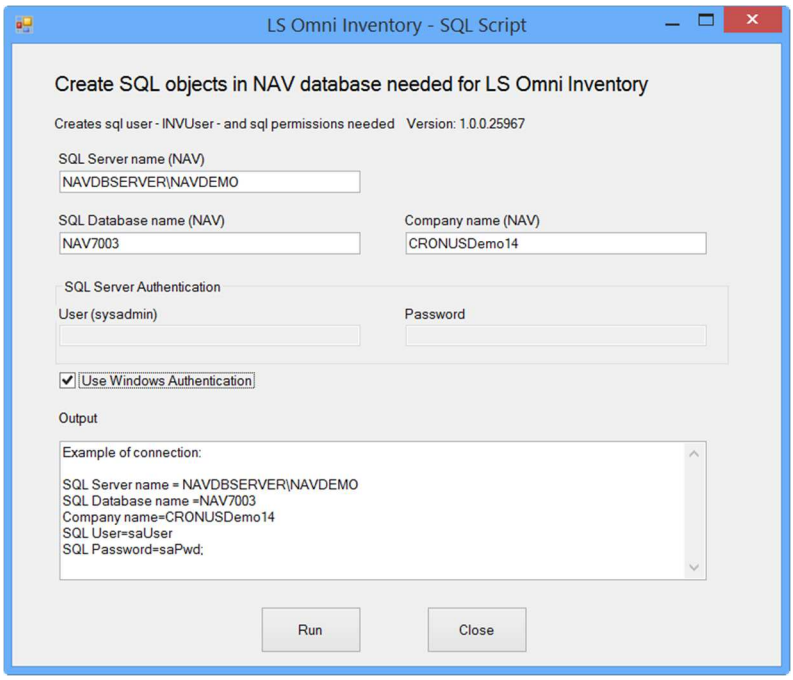
3.1 Create SQL Server permissions in the NAV database

Data needs to be replicated from NAV to the mobile device where it is stored in a local database. This requires select permissions on NAV SQL Server tables.

Run the ***LSOmni.Inventory.NAVDBSetup.exe*** installation program and enter appropriate values in the text boxes. The Data Source and Initial Catalog is equivalent to those in a SQL Server connection string.

- NAV Server (Data Source) : ***NAMEOFNAVDATABASESERVER***(\NameOfDatabaseInstance)
- NAV Database (Inital Catalog) : ***NAMEOFNAVDATABASE***
- NAV Company Name : **NAMEOFNAVCOMPANYNAME**

Use either SQL Server Authentication (user / password) or Windows Authentication to connect to SQL Server. This user must have sysadmin privileges in SQL Server since a new SQL Server login/user is created and permissions are granted to SQL Server tables in the NAV database.



NOTE: This setup program creates a SQL Server login/user called **INVUser** (pwd: INVUser) that is used to connect to NAV and replicate data. This is done to simplify the installation process. This user can be replaced or if integrated security is preferred in the sql connection string.

LSOmni.Inventory.NAVDBSetup.exe can be run multiple times. However there is no "uninstall" so the SQL Server INVUser must be manually removed from the database.

3.2 INVService – a WCF web service running under IIS

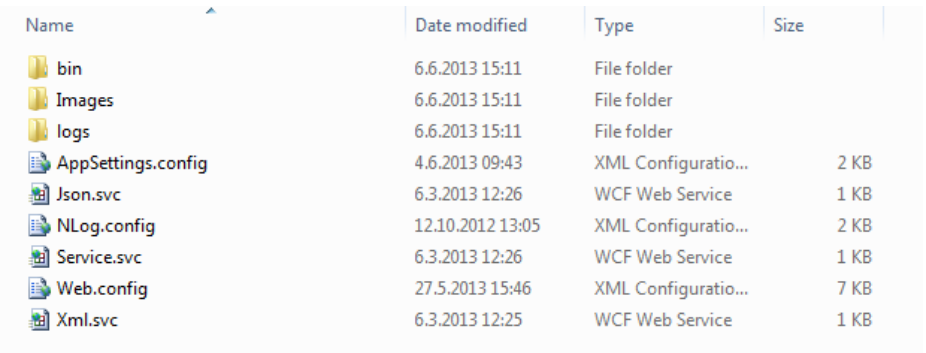
Before the NVService can be installed, IIS and other required Windows features must be enabled. See appendix Installing IIS and WCF for detailed information on how to install IIS 7.x/8 on a Windows machine.

The INVService is a WCF web service running under IIS 7.x /8. The iOS device calls the INVService which calls the NAV server. The INVService has no business logic; it simply passes the data between the mobile device and NAV. The INVService has two ways to communicate with NAV.

- Calls NAV web services for operations like item lookup or retrieving documents
  - A NAV user and password is read from the appsettings.config file and added as credentials to each NAV web service requests. This NAV user must have permission in NAV to execute web service requests.
- Calls NAV SQL Server directly to replicate data from NAV to mobile device
  - This uses the INVUser SQL user to connect to the NAV SQL Server. The SQL connection string is n the appsettings.config file.

The WCF INVService is installed by running the ***LSOmni.Inventory.WcfServiceSetup.msi*** installer package that walks through the complete installation process. See step through guide in appendix LS Omni Mobile INVService Installer package.

Once deployed, open the \INVService folder in Windows Explorer and verify the values used in the **AppSettings.config** file.



Name	Date modified	Type	Size
bin	6.6.2013 15:11	File folder	
Images	6.6.2013 15:11	File folder	
logs	6.6.2013 15:11	File folder	
AppSettings.config	4.6.2013 09:43	XML Configuratio...	2 KB
Json.svc	6.3.2013 12:26	WCF Web Service	1 KB
NLog.config	12.10.2012 13:05	XML Configuratio...	2 KB
Service.svc	6.3.2013 12:26	WCF Web Service	1 KB
Web.config	27.5.2013 15:46	XML Configuratio...	7 KB
Xml.svc	6.3.2013 12:25	WCF Web Service	1 KB

The AppSettings.config file contains all the connection settings used the INVService. The INVService web service needs to be restarted if this file is modified.

```
<appSettings>
  <!-- INVConnectionString, sql server connection string -->
  <add key="INVConnectionString" value="Data Source=NAVDBSERVERNAME(/NavDbI
nstance);Initial Catalog=NAVDBNAME;
          Persist Security Info=True;User ID=INVUser;Password=INVUser; NAVC
ompanyName=NAVCOMPANYNAME(ie.Cronus)"/>

  <add key="BOConnection.Nav.INV.Url" value="http://NavWebServiceName:PORTN
O
(ie.7047)/NAVSERVICEINSTANCE(ie.DynamicsNAV70)/WS/NAVCOMPANYNAME(ie.CRONUS)
/Codeunit/RetailWebServices"/>
  <add key="BOConnection.Nav.INV.UserName" value="NAVWINDOWSUSERNAME"/>
  <add key="BOConnection.Nav.INV.Password" value="NAVUSERPASSWORD"/>

… more values …
</appSettings>
```

Text in bold and italics should be modified according to LS Retail NAV environment. Parentheses are just for clarification and should be removed.

- „**INVConnectionString**“ is the sql server connection string used to retrieve data that is replicated.
- „**BOConnection.Nav.INV.Url**“ is the url to the NAV web service
- „**BOConnection.Nav.INV.UserName**“ is the windows user used to connect to the NAV web service, has role SUPER in NAV and is dbo or has equivalent rights in NAV database.
- „**BOConnection.Nav.INV.Password**“ is the above BOConnection.Nav.INV.User password. It can be encrypted with the MobileInventoryGenerator.exe application that is a part of this release

The web.config file does not need to be changed unless SSL is enabled.

To validate that the INVService web service is up and running, open a browser (Chrome/Firefox) with the URL to appropriate machine to these links.

- http://localhost/invservice/json.svc/ping returns “PONG from DB” if web service is up and running and has privilege to access NAV Web services (connects to NAV through web service and returns a string from NAV database)
- http://localhost/invservice/json.svc/version returns “1.1.0.*” if web service is up and running (does not call NAV only INV web service)

All errors are logged to INVService root folder\logs\logfile.txt. ***Please check this file for detailed error messages.***

Please refer to “General setup for Mobile in NAV” chapter for NAV Web Services Setup.

4 Installation of the Mobile Inventory application

Currently the LS Omni Mobile Inventory application runs on Windows Mobile 6.1 (and above), Windows CE 6.0 (and above) and requires .Net Compact Framework 3.5.

***Disclaimer**: Windows CE is an OEM-only, embedded operation system. LS Retail cannot guarantee that the Mobile Inventory application will run on all Windows CE based mobile devices.*

LS Retail has tested Windows CE on Motorola MC 2100

4.1 Installation on mobile device

It is strongly recommended that you test the connection to the NAV web service using the test programs provided before installing the Mobile Inventory application. See appendix B.

Install the mobile inventory application on the mobile device.

- Install from the mobile device (cabinet installation).
  Make sure you have *Windows Mobile Device Center* installed.
  Copy the LSRetailMobileInventory.CAB cabinet file to the device and run it from the device (by clicking the cab file), and follow the on screen instructions. Safely click OK to use default settings.
- .Net Compact Framework 3.5 is required. If it is not already installed on the mobile device then install the NETCFv35.wm.armv4i.cab cabinet file (NETCFv35.wce.armv4.cab if using Windows CE).

The installation creates a shortcut under Programs and can be uninstalled at any point.

By default the application is installed under "Program Files\LSRetailMobileInventory".

This folder contains a configuration file called **app.config**. This config file must be modified before running the application. The values shown below must be modified.

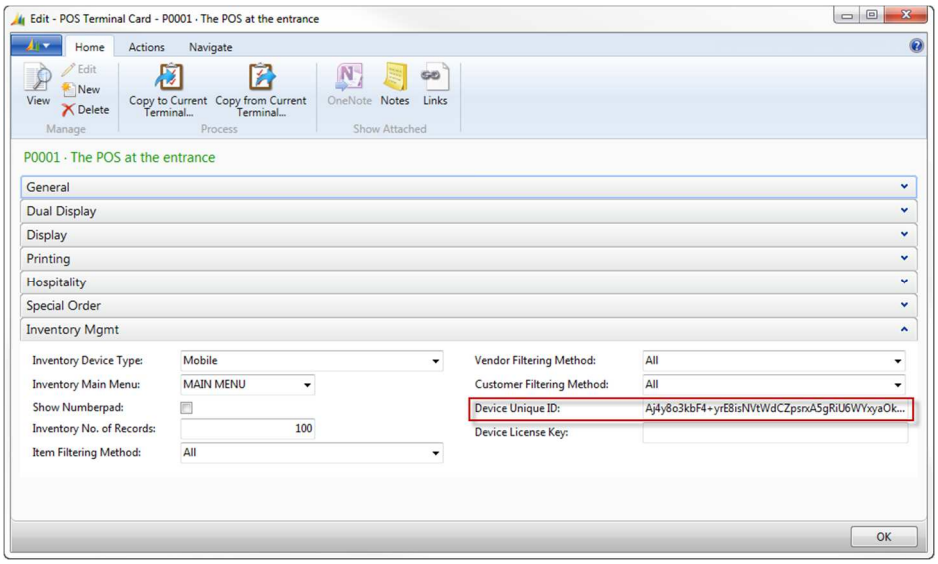
```
<add key="ApplicationTitle" value="LS Retail Mobile Inventory" />
<add key="Terminal" value="P0001"/>
<add key="Store" value="S0001"/>
<!-- LogLevel (logs to logs\logfile.txt). level 1=trace, 2=errors only -->
<add key="LogLevel" value="1"/>

<!-- IIS Inventory web service,  -->
<add key="INVService.Url" value="http://172.22.27.274/INVService/Json.svc/" />
<add key="INVService.TimeoutInSec" value="10"/>
```

4.2 Activation

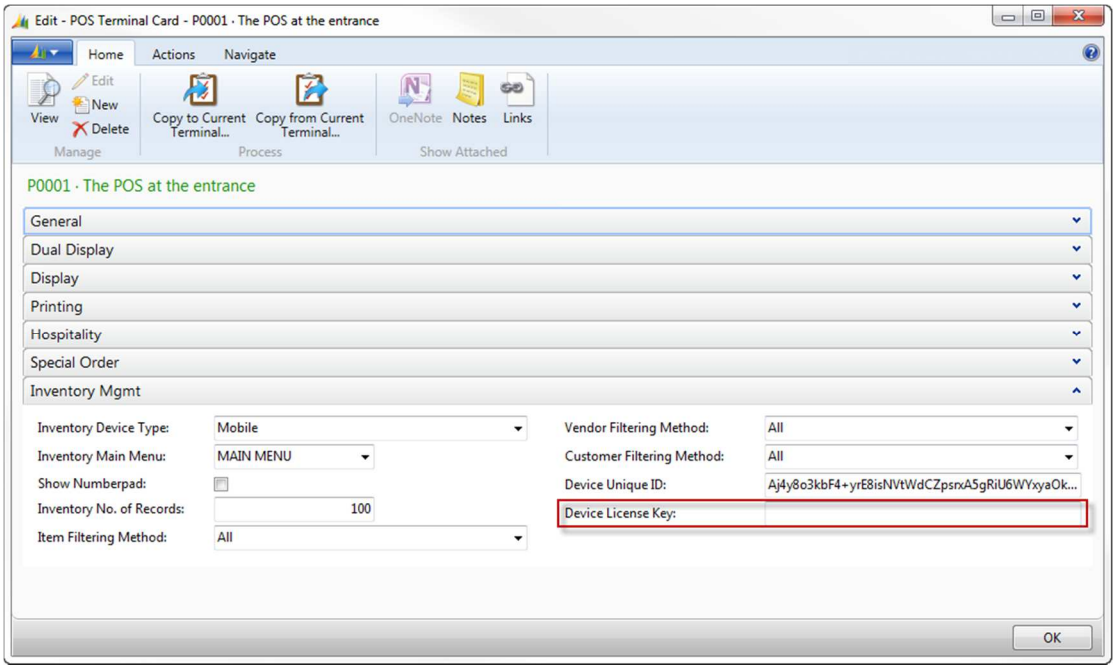
The mobile application requires a license issued by NAV. However a 30 day trial license is generated on the mobile device if no license is issued by NAV.

Each mobile device is licensed separately and therefore needs a unique license key applicable only to that particular device. To issue a license for a mobile device the following procedure must be carried out:

1) Successfully connect the Mobile Inventory application to the NAV Web Services.
2) Start the Mobile Inventory application on the mobile device.
3) Navigate to the POS Terminal Card in NAV and verify that the *Device Unique ID* has been successfully imported from the mobile device.



4) Forward the *Device Unique ID* to LS Retail Support to receive a Device License Key which is then entered or pasted into the *Device License Key* field.

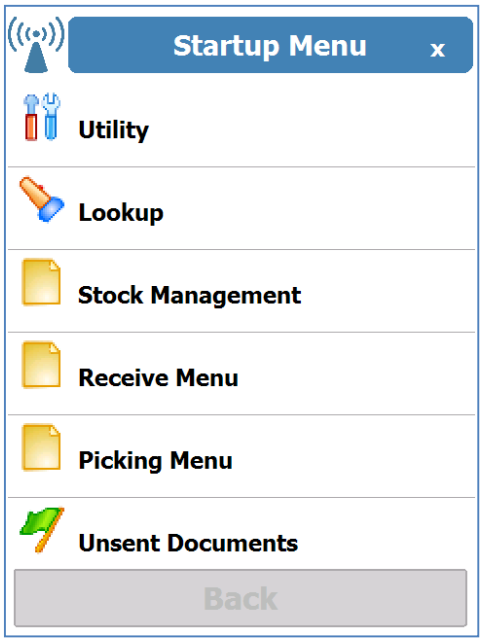


5) Restart the Mobile Inventory application on the device.
6) Verify that the application has been fully licensed. This information is visible on the logon screen of the application.

5 Functionality

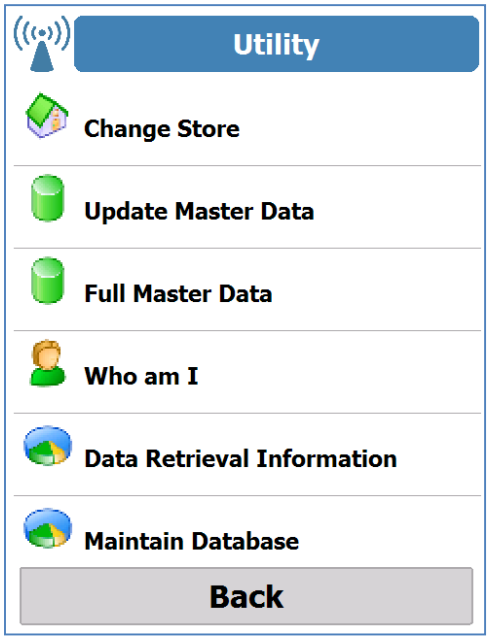
The LS Retail Mobile Inventory application is fully configured in LS Retail NAV 7.0. At startup the application gets its setup configuration from the NAV web service. This includes all the menus available to the user, as well as data about the terminal, store and staff. This data along with customer, vendor and item information is stored locally on the device. The mobile application has multilanguage support.

5.1 Startup Menu



The first menu displayed after login, the Startup Menu, leads to all the main features/actions of the Mobile Inventory application. ***Note that menu headers and menu lines used in this document are based on the LS Retail demo data.*** The Startup Menu and its menu lines are all created in NAV and can vary between NAV mobile setups.

5.2 Utility



The Utility Menu includes the following actions:

- **Change Store**
  The store used by the mobile application is read from the app.config file. If the mobile device is moved between stores then simply select the new store from a list of available stores. Note that the local database deletes all the data related to the old store.
- **Update Master Data**
  Updates only the changes made in NAV since the last update of the master data.
- **Full Master Data**
  Retrieves all the master data from NAV. The database is first cleared and then the data is stored, no matter whether this is for the first time or replacing existing data. ***Note this action must be done before using the application.***
- **Who am I**
  Shows the information about the user and mobile device.
- **Data Retrieval Information**
  A list of data that has been retrieved from NAV and stored in the local database.
- **Maintain Database**
  The local database is compacted and old log data expunged.

5.3 Lookup

All lookups require online connection to NAV. If the mobile application is offline, a notification is displayed showing that the NAV server connection is not available. At all times - in any menu - it is possible to check if you are online in the left hand corner with the red X indicating that you are not connected to NAV. The X on the right hand corner allows you to log off.

5.3.1 Scan or local search

Item Lookup

It is possible to look items up with a barcode, either by scanning or entering the barcode and get information on the item, including the number of items that correspond to this barcode. The search button on the right hand side opens the Item Search where the user can search by item Id or item description.

Item Lookup by color (size known)

Same as item look up but for items with variants it is possible to get all colors for an item in certain size. An example might be when item 40020 Linda skirt is scanned in by the barcode for pink in size 36. Here information on all items in size 36 would be displayed, for all available colors, showing the number of items in each color for this size.

Item Lookup by size (color known)

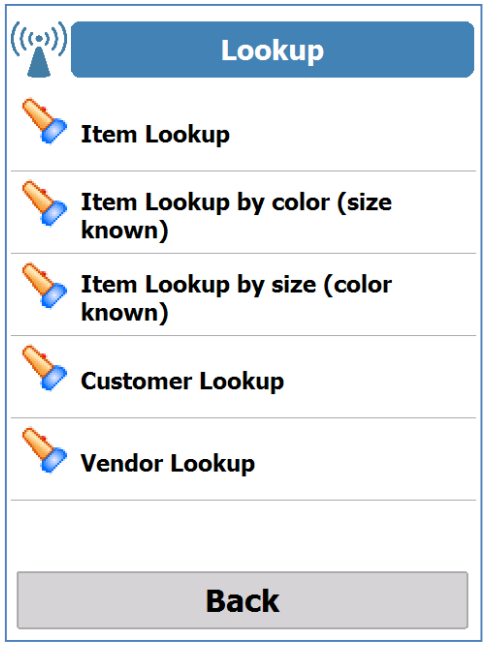
Same as item lookup but for items with variants it is possible to get all sizes for an item in a certain color. An example might be when item 40020 Linda skirt is scanned in by the barcode for pink in size 36. Here information on all items in pink would be displayed, for all available sizes, showing the number of items in each size for this color.

Customer Lookup

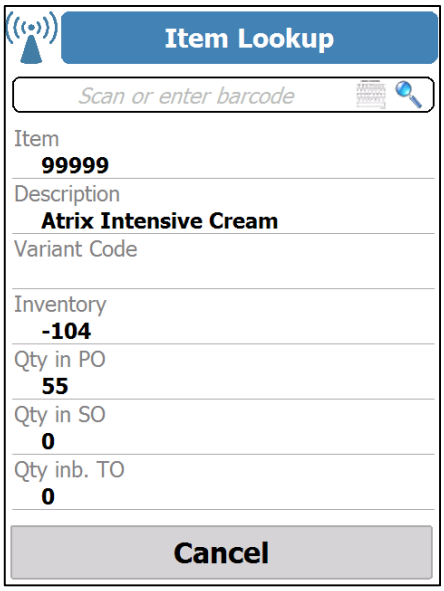
It is possible to look up a customer with a barcode, either by scanning or entering the barcode and get information on the customer. The search button on the right hand side opens the Customer Search where the user can search by customer name.

Vendor Lookup

It is possible to look up a vendor with a barcode, either by scanning or entering the barcode and get information on the vendor. The search button on the right hand side opens the Vendor Search where the user can search by vendor name.



To do any lookups the user must be online as all lookup requests are sent to the NAV web service and NAV decides what data is returned and displayed on the mobile screen. The item lookup above has three views of the item, each returning different data based on the item card view used.



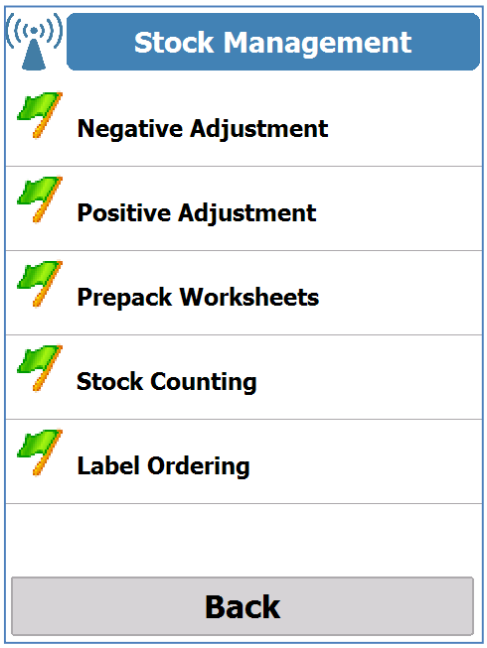
5.4 Worksheets

Unlike the Lookup functionality, worksheets can be processed offline and sent to NAV when the mobile device is online. Worksheets are always saved in the local database so the user can send the worksheet at any point in time. The main inventory management functions that involve worksheets are the following:

- Purchasing
- Purchase Return
- Stock Transfer
- Stock Request
- Stock Counting
- Positive Adjustment
- Negative Adjustment
- Pre-pack
- Label Ordering

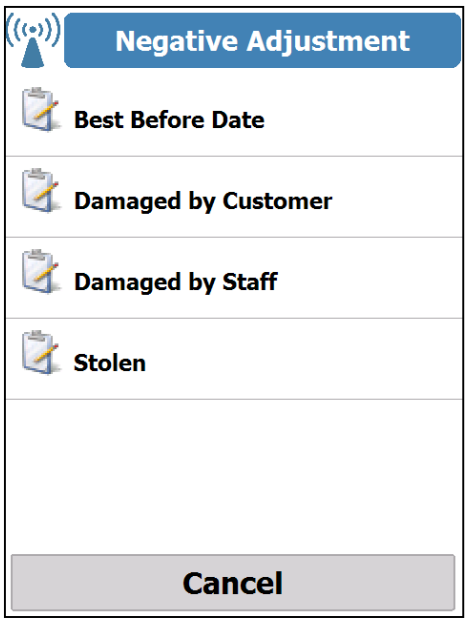
5.4.1 Stock Management

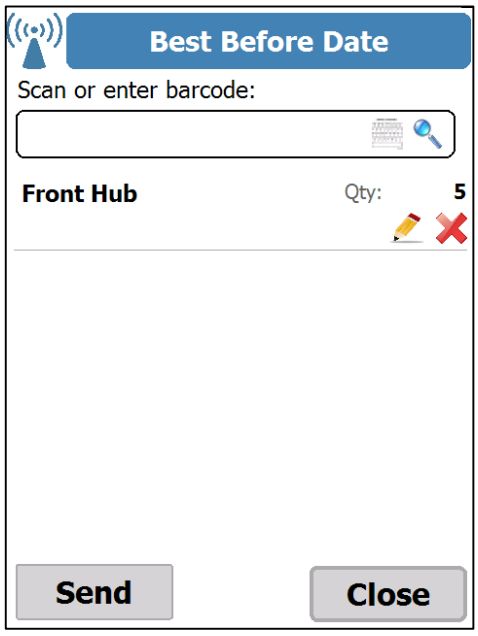
Our Stock Management menu contains the worksheets and it has sub-menus with more granular worksheet functionality.



5.4.2 Negative Adjustment

The Negative Adjustment worksheet is an example of a worksheet that includes several adjustments that may be needed over time for stock management, such as adjustments because of damaged goods, theft or other causes that will change the number of items in stock.


Negative Adjustment
Best Before Date
Damaged by Customer
Damaged by Staff
Stolen
Cancel


Best Before Date
Scan or enter barcode:
Front Hub
Qty:
5
Send
Close

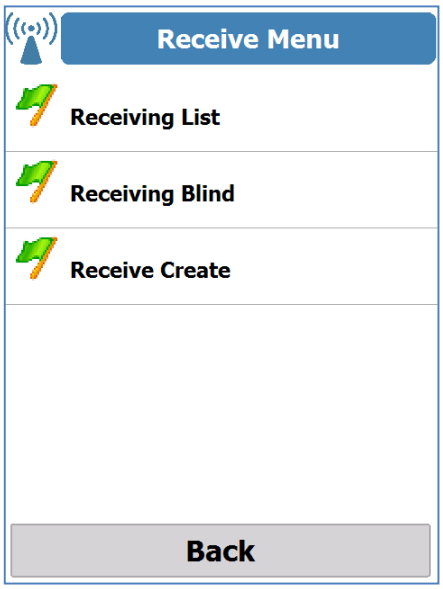
5.5 Documents

The documents are grouped under two menus, Receive and Pick menus. Both of them have documents grouped under the list, blind and create menus.

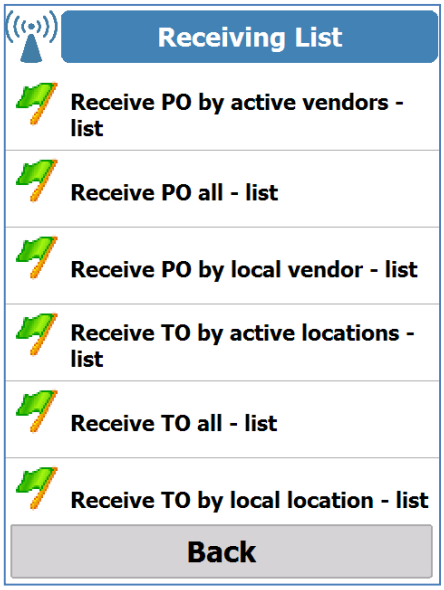
A list document shows the user what items are in the document, whereas the blind and create documents do not show the user any items. A warning is given to the user when trying to send a blind document and the quantity entered does not match with the quantity in document, and at this point a recount can be requested.

To identify a document, the user must go through a selection process where filtering can be done on location, customer, vendor or active documents. Or the user can simply scan items to identify the document. During the selection process the mobile device must be online but once the document has been downloaded the user can work offline until the document is ready to be sent to NAV. The document can be saved to a local database and sent at a later date.

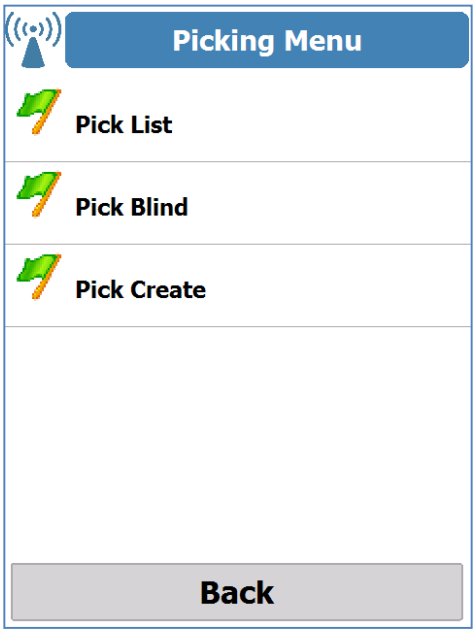
5.5.1 Receive Menu



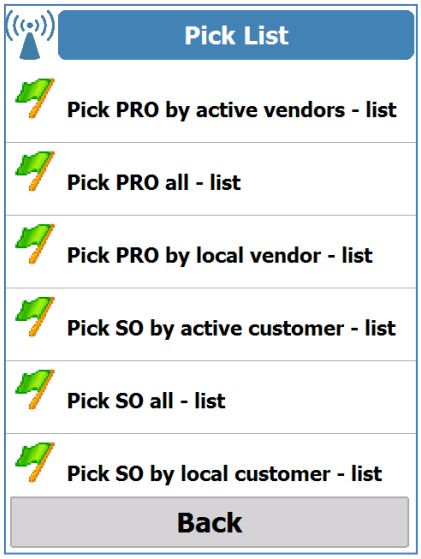
The full list for the Receive Menu and its sub-menus can be found in appendix A.



5.5.2 Picking Menu



The full list for the Pick Menu and its sub-menus can be found in Appendix A.


Pick List
Pick PRO by active vendors - list
Pick PRO all - list
Pick PRO by local vendor - list
Pick SO by active customer - list
Pick SO all - list
Pick SO by local customer - list
Back

6 NAV configuration for mobile inventory

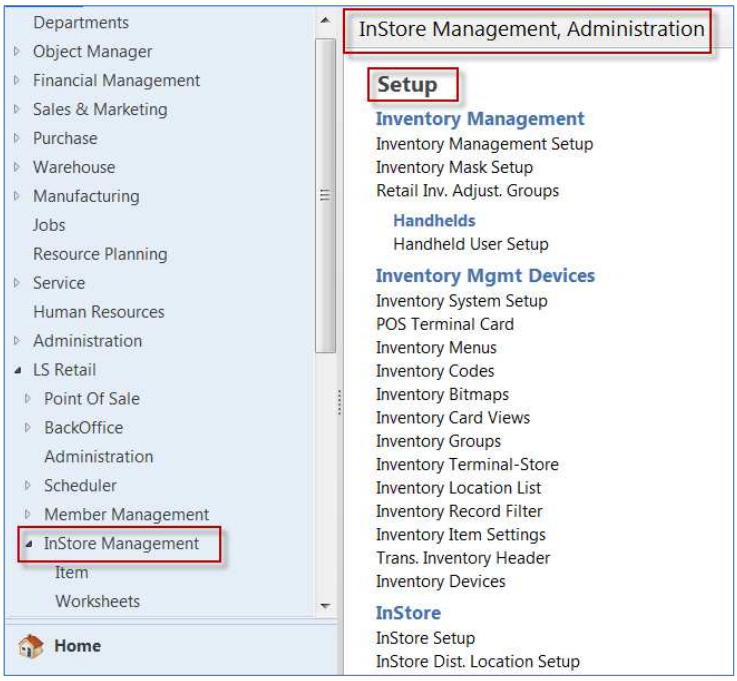
Most of the mobile device setup is done within NAV. This includes menus, menu items and data communication. NAV web services must be installed, running and configured to communicate with the relevant NAV database, in order to use the Inventory Management on the mobile device. The web services setup is not described in this document, as it is part of standard NAV.

The three following chapters describe the setup needed in LS Retail NAV for Mobile Inventory. This chapter deals with General setup and includes web services related to Inventory Processes, as well as Inventory general setup (for example how to set up the menus shown on the mobile device). The Terminal Setup chapter describes what terminal setup is needed. Final chapter is about staff setup. Appendix A shows Demo data for menus and other data that is not part of the default data.

The following setup must be done in order to use a mobile for Inventory Management:

- Inventory System Setup
- Inventory Menus
- Inventory Codes
- Inventory Card Views
- Inventory Terminal-Store
- Inventory Location List
- Web services
- Terminal Setup
- Staff setup

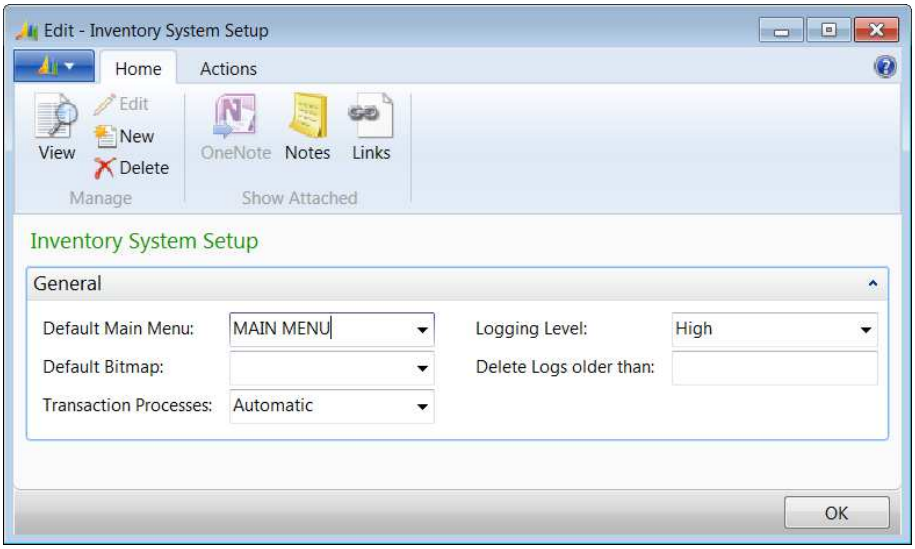
The setup can be found under "**LS Retail – InStore Mgmt**"



6.1 General setup for Mobile in NAV

The following screenshots contain the same as data as in the Demo Database that comes with LS Retail NAV 7.0, and is example data.

Inventory System Setup



Before a default menu can be defined here it must already be available in the Menu table, see the Inventory Menus chapter below.

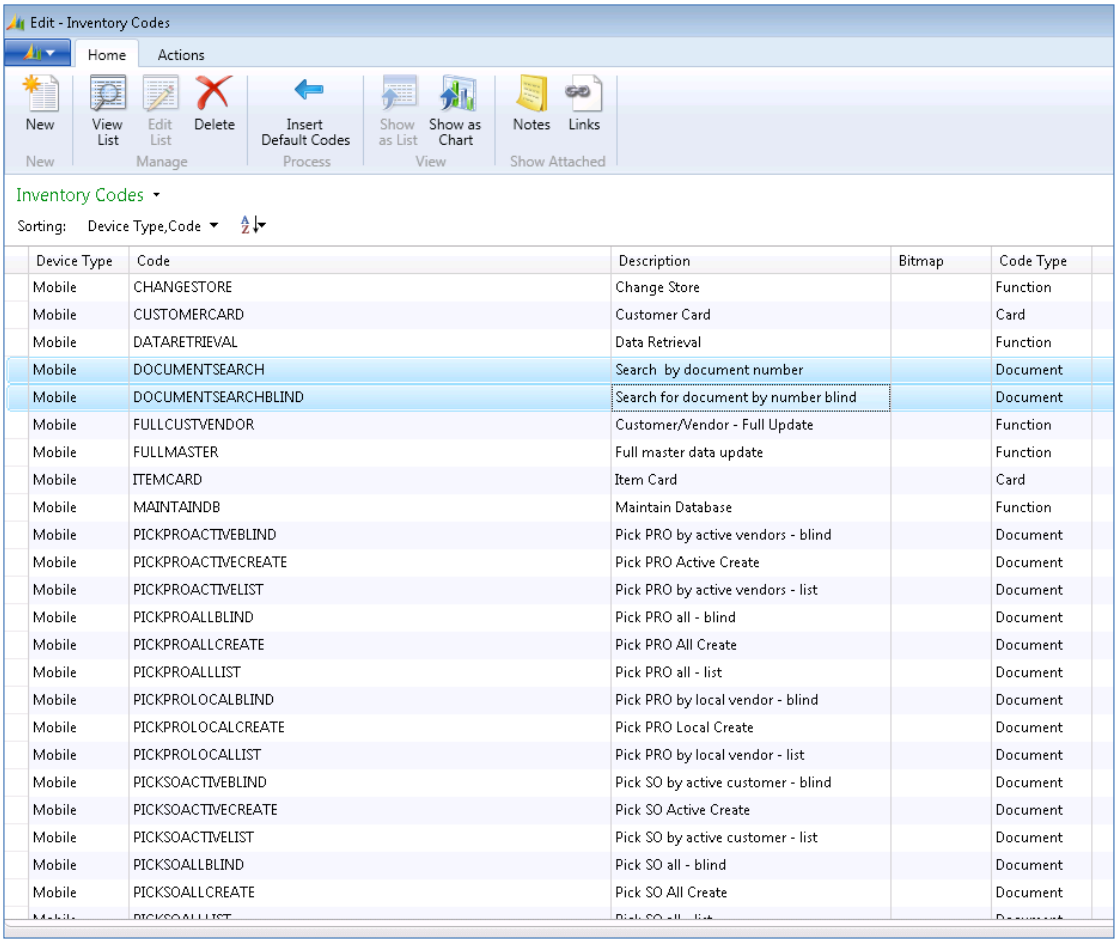
Default Main Menu and **Default Bitmap** can be defined here and used if they are not defined in the terminal setup. There are icons delivered with the Inventory Mobile solution that can be used, however these can be modified to any icon. LS Retail is not responsible for any copyright infringement in using other icons.

Transaction Processes. Automatic is set by default. Automatic means that all transactions coming from the mobile device are processed automatically from the Inventory Transaction table into relevant tables such as worksheets, receiving documents and others. None means that the transactions are not processes but it can be done manually from the Inventory Transaction form.

Logging Level allows the user to define various levels of logging, the log file can be viewed on the Inventory Devices form.

Inventory Codes

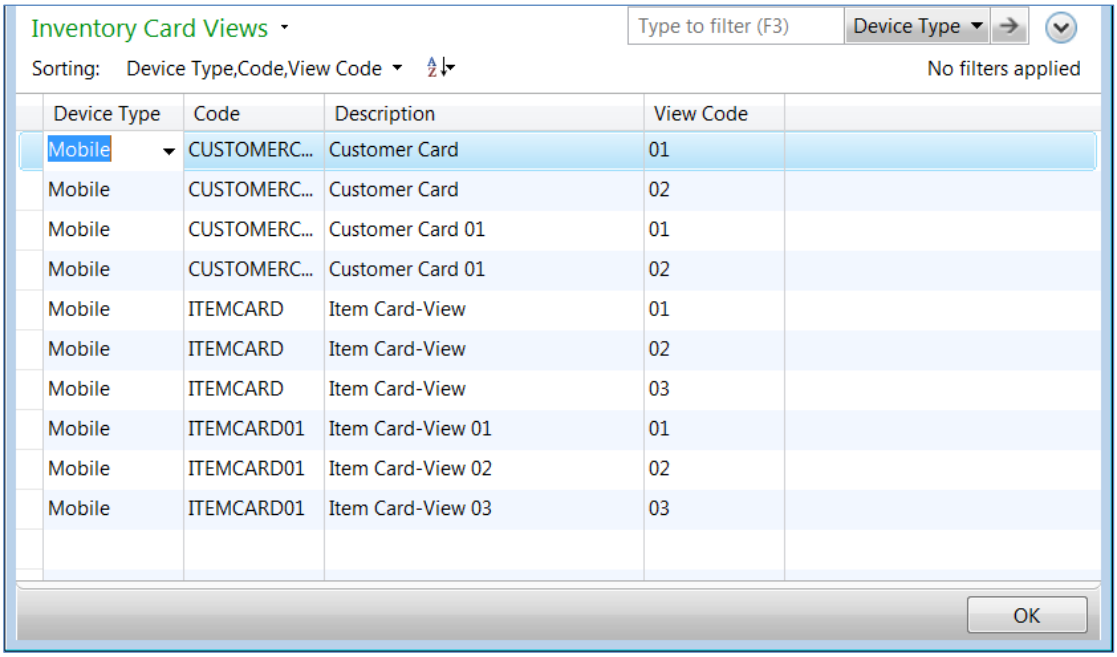
This table holds the definition of the code or operation behind the menu items. The codes can be set up from the function button. See also Appendix A, which shows the setup in the Demo database.



Device Type	Code	Description	Bitmap	Code Type
Mobile	CHANGESTORE	Change Store		Function
Mobile	CUSTOMERCARD	Customer Card		Card
Mobile	DATARETRIEVAL	Data Retrieval		Function
Mobile	DOCUMENTSEARCH	Search by document number		Document
Mobile	DOCUMENTSEARCHBLIND	Search for document by number blind		Document
Mobile	FULLCUSTVENDOR	Customer/Vendor - Full Update		Function
Mobile	FULLMASTER	Full master data update		Function
Mobile	ITEMCARD	Item Card		Card
Mobile	MAINTAINDB	Maintain Database		Function
Mobile	PICKPROACTIVEBLIND	Pick PRO by active vendors - blind		Document
Mobile	PICKPROACTIVECREATE	Pick PRO Active Create		Document
Mobile	PICKPROACTIVELIST	Pick PRO by active vendors - list		Document
Mobile	PICKPROALLBLIND	Pick PRO all - blind		Document
Mobile	PICKPROALLCREATE	Pick PRO All Create		Document
Mobile	PICKPROALLLIST	Pick PRO all - list		Document
Mobile	PICKPROLOCALBLIND	Pick PRO by local vendor - blind		Document
Mobile	PICKPROLOCALCREATE	Pick PRO Local Create		Document
Mobile	PICKPROLOCALLIST	Pick PRO by local vendor - list		Document
Mobile	PICKSOACTIVEBLIND	Pick SO by active customer - blind		Document
Mobile	PICKSOACTIVECREATE	Pick SO Active Create		Document
Mobile	PICKSOACTIVELIST	Pick SO by active customer - list		Document
Mobile	PICKSOALLBLIND	Pick SO all - blind		Document
Mobile	PICKSOALLCREATE	Pick SO All Create		Document

Inventory Card Views

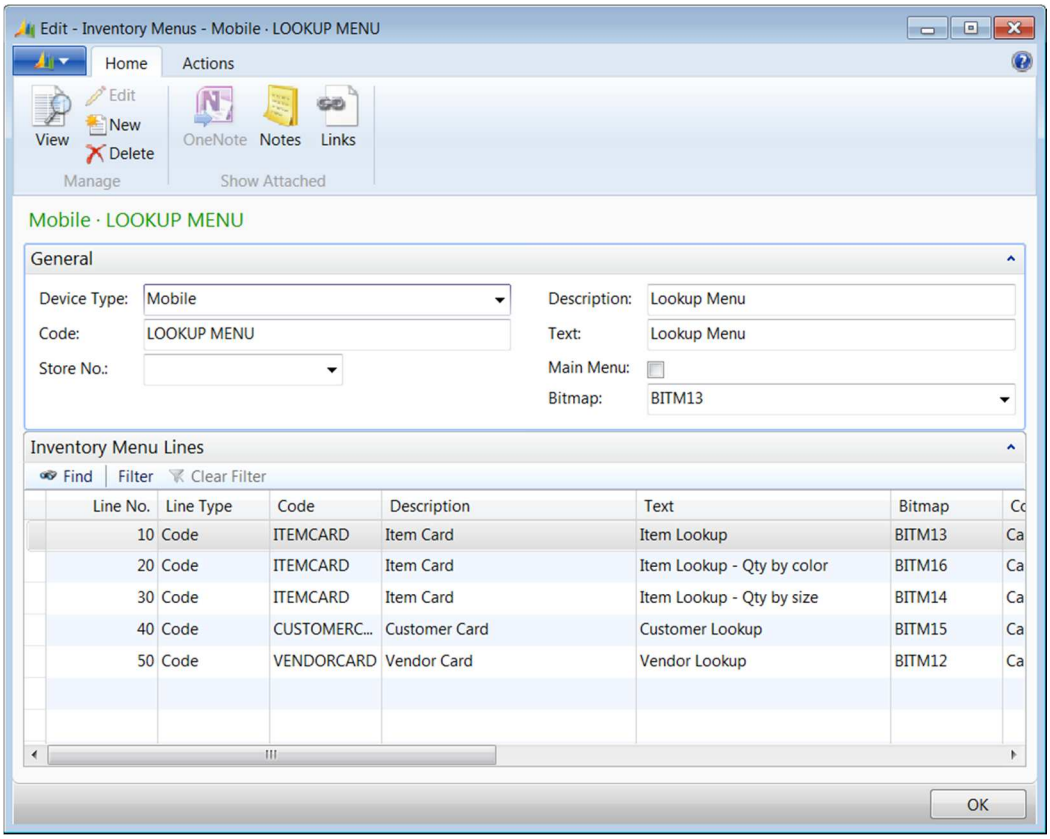
Inventory Card Views is the setup for the lookup on the Mobile device. It does not come with the default data but can easily be copied from the NAV Demo Database, as the following screenshot shows.



Inventory Menus

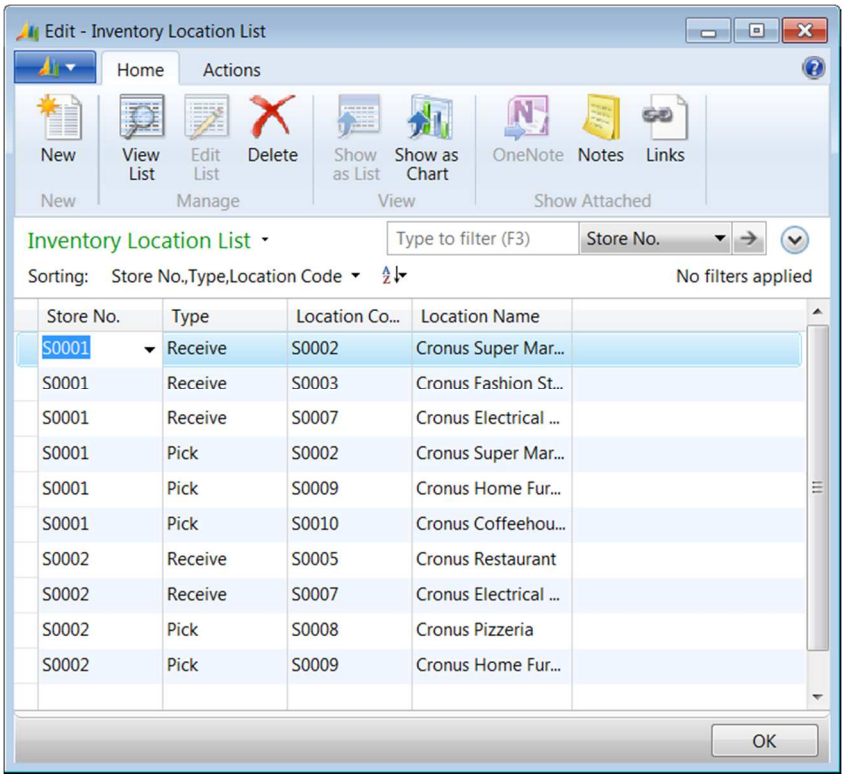
All inventory menus and menu items for the mobile devices are set up and configured here. A menu can contain either another menu or menu item (Code). Menus can be used down to 5 levels. The menu therefore has a tree structure.

The menus are not part of default data. However they are a vital part of the setup and Mobile Inventory will not work without it. The menus can be copied from the NAV 7.0 Demo Database, the same way as Inventory Card Items, or set up according to Appendix A, that shows all menus and menu items included in the LS Retail NAV 7.0. Demo Data.



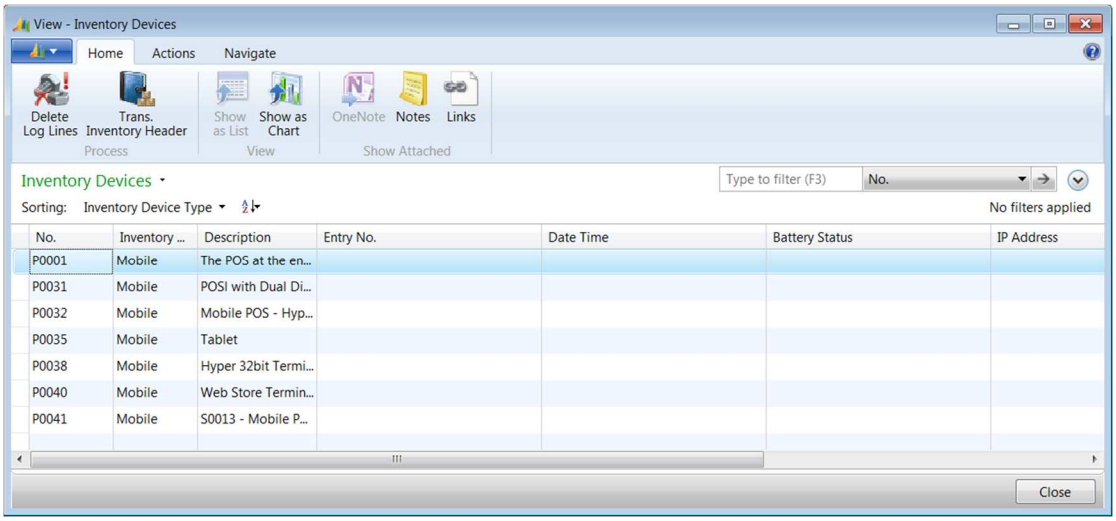
Inventory Location List

The Inventory Location list is used for transfer orders when transferring, receiving or picking locally. It displays the list the user can select from.



Inventory Devices

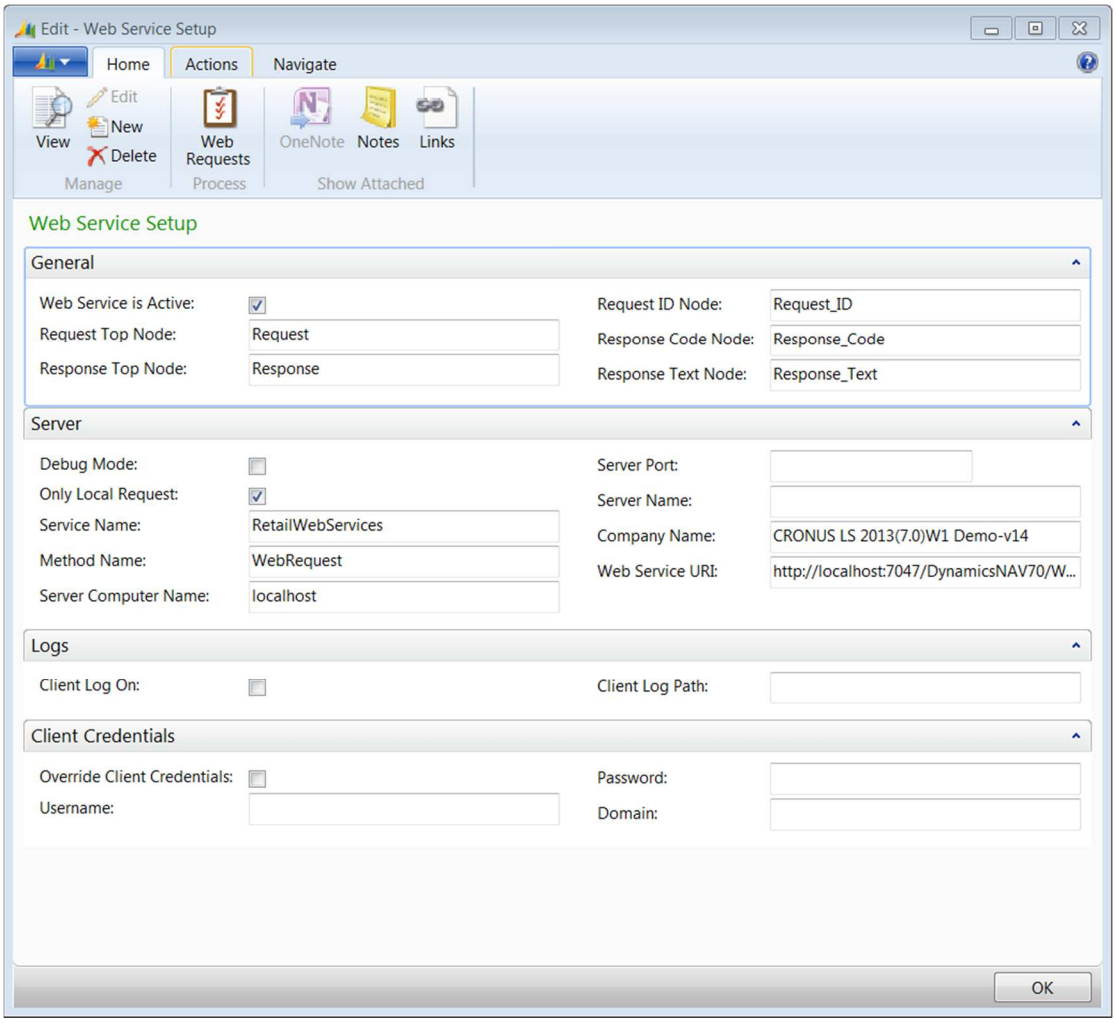
Shows a list of inventory mobile devices that have already been set up for terminals on all the terminal cards. When a mobile device is connected various information are filled in each time such as Entry Number, Date and Time, Battery status and so on. The functions buttons includes the log files and other functions.



Web service setup

LS Retail, Administration, Web Service, Web Service Setup

Values on the Server tab need to be configured according to each database setup.

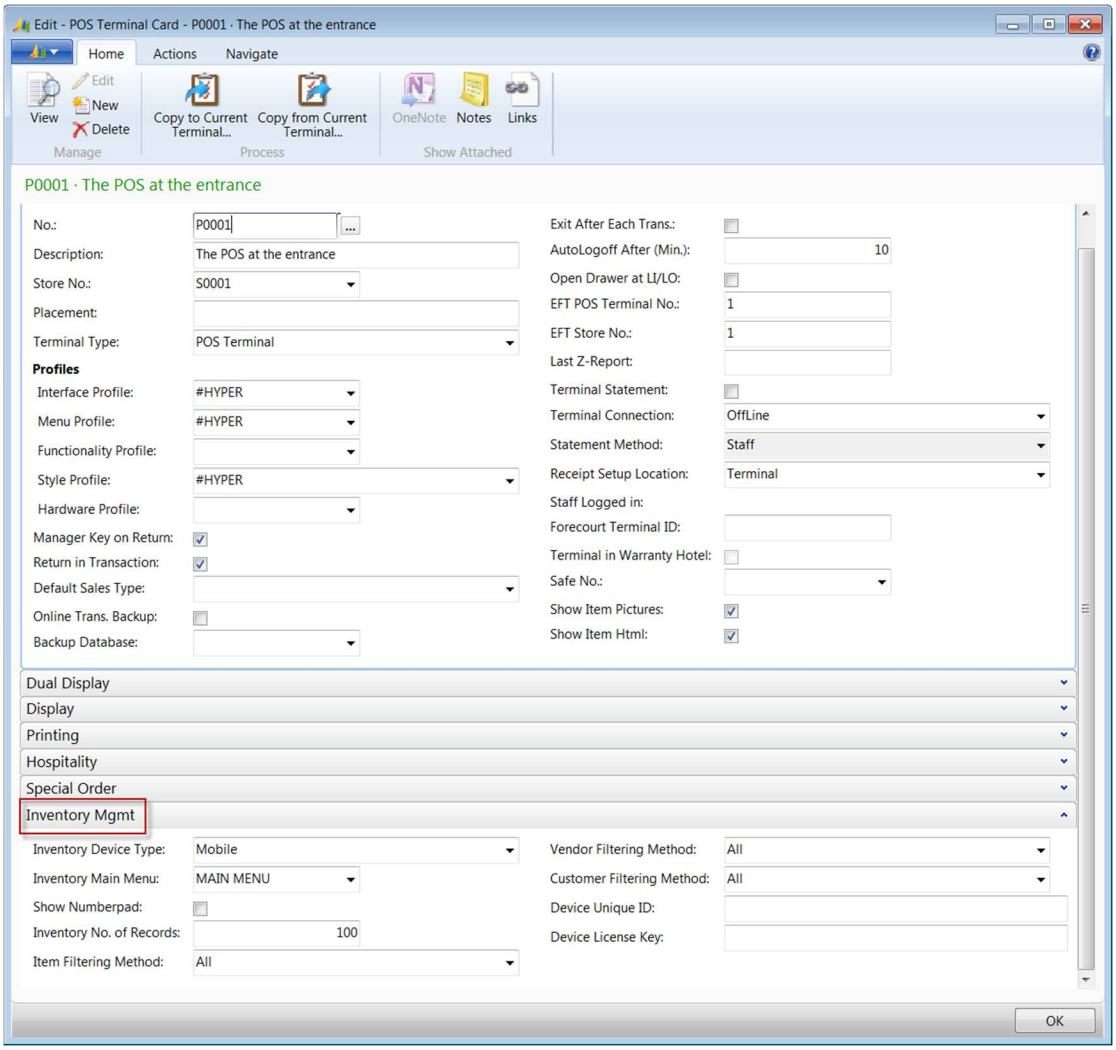


Detailed documentation on web services setup can be found on the partner portal .

6.2 Terminal

Terminal

Setup for the terminal is done at the **LS Retail, Point of Sale, POS, Administration, General, Terminals**, select a terminal or create a new one pressing the button **New**.



Inventory Device Type. Terminal that is to be used for Mobile device must have type Mobile.

Inventory Main Menu. The setup allows various main menus for each terminal. If nothing is selected here that main menu from Inventory System Setup is used.

Show Numberpad. When using mobile device with numpad, showing numpad is not needed, otherwise ensure this is checked.

The next 4 fields concern loading master files to mobile device:

Inventory No. of records, definition of number of records to avoid timeout.

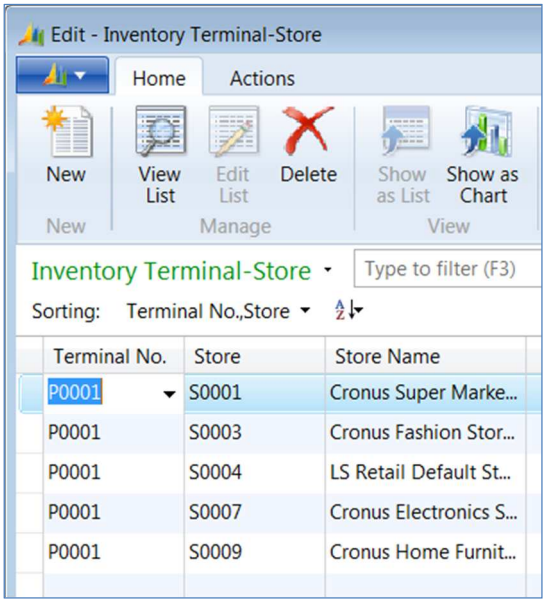
For next three fields: **Item Filtering Method, Vendor Filtering Method** and **Customer Filtering Method**, user can set various filter when downloading the master data to the mobile device.

Device Unique ID is sent from the mobile device and is 44 characters long.

Device License Key is also sent from the mobile device.

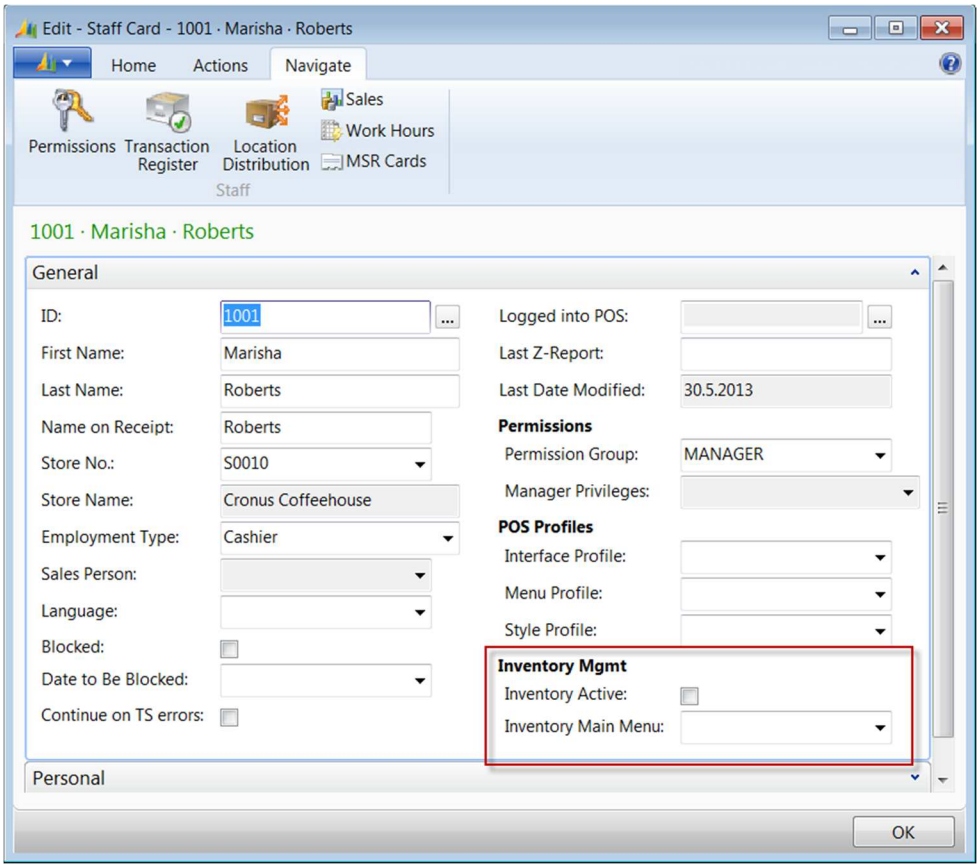
Inventory Terminal-Store

When a user wishes to change the mobile device to another store, this is the list shown on the mobile device, where he can select stores from.



6.3 Staff

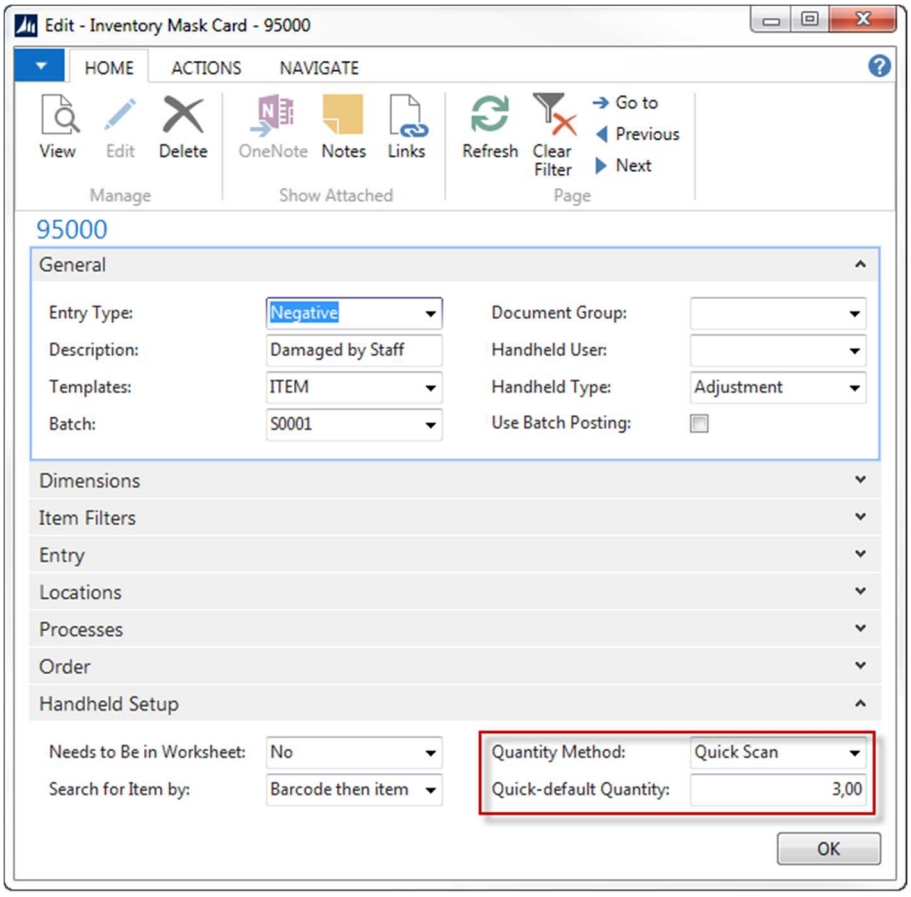
Staff members can be set up if they are using mobile inventory, if Inventory is active and inventory main menu for the staff. If no main menu is defined here, the information is taken from Terminal Setup or Inventory System Setup.



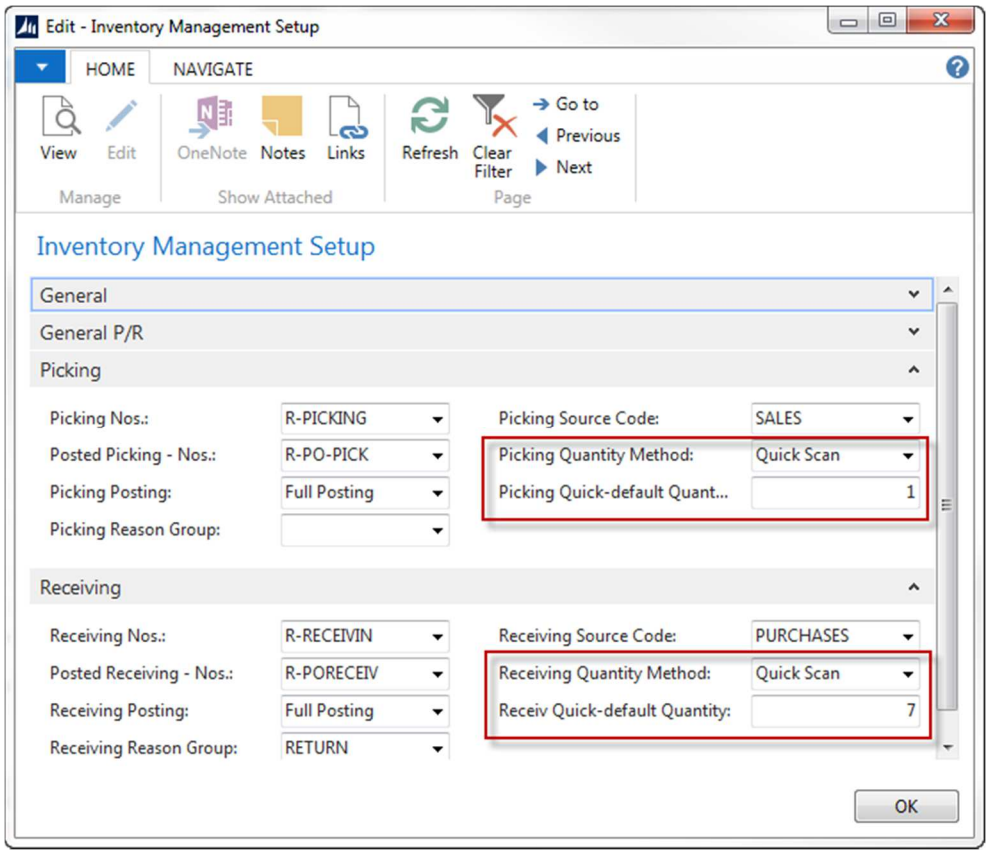
6.4 Quick Scan

"Quick scan" values can be defined per worksheet as well as for picking and receiving document.

The quick scan value for the worksheets is assigned on the **Handheld Setup** section of the desired **Inventory Mask Card** by setting the **Quantity Method: Quick Scan** and the **Quick-default Quantity**.



The quick scan value for picking and receiving documents is assigned on the **Picking** and **Receiving** sections of the **Inventory Management Setup** page by setting the **Quantity Method: Quick Scan** and the **Quick-default Quantity**.



In Navision 2013, all modified values are committed when the page is closed. The hand held unit will not pick up the new value(s) if the page is left open.

Appendix A

The below table contains a list of the actions handled on the handheld device

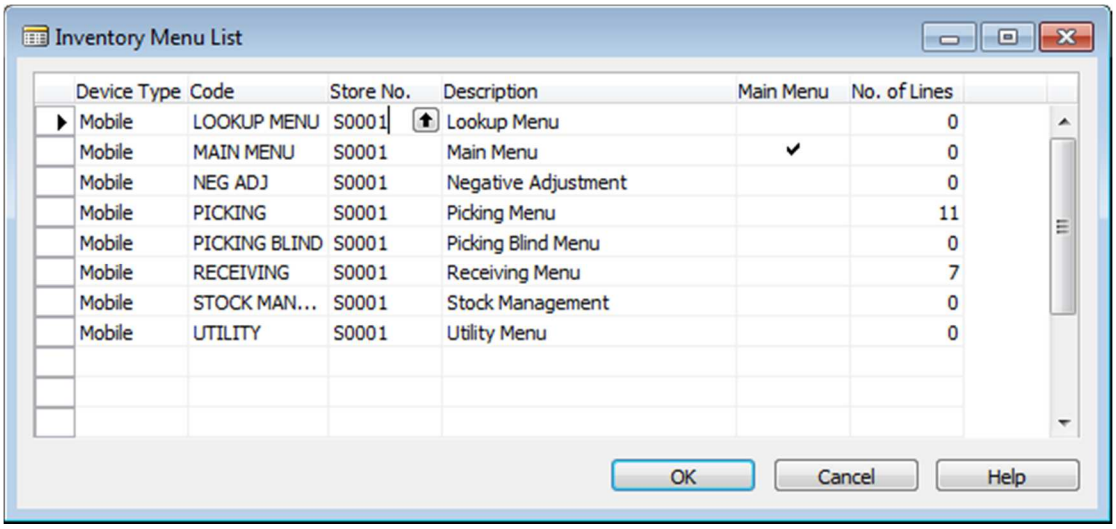
Inventory Codes

Device Type	Code	Description	Bitmap	Code Type
Mobile	CHANGESTORE	Change Store		Function
Mobile	CUSTOMERCARD	Customer Card		Card
Mobile	DATARETRIEVAL	Data Retrieval		Function
Mobile	DOCUMENTSEARCH	Search by document number		Document
Mobile	DOCUMENTSEARCHBLIND	Search by document number blind		Document
Mobile	FULLMASTER	Full master data update		Function
Mobile	ITEMCARD	Item Card		Card
Mobile	MAINTAINDB	Maintain Database		Function
Mobile	PICKPROACTIVEBLIND	Pick PRO by active vendors - blind		Document
Mobile	PICKPROACTIVECREATE	Pick PRO Active Create		Document
Mobile	PICKPROACTIVELIST	Pick PRO by active vendors - list		Document
Mobile	PICKPROALLBLIND	Pick PRO all - blind		Document
Mobile	PICKPROALLCREATE	Pick PRO All Create		Document
Mobile	PICKPROALLLIST	Pick PRO all - list		Document
Mobile	PICKPROLOCALBLIND	Pick PRO by local vendor - blind		Document
Mobile	PICKPROLOCALCREATE	Pick PRO Local Create		Document
Mobile	PICKPROLOCALLIST	Pick PRO by local vendor - list		Document
Mobile	PICKSOACTIVEBLIND	Pick SO by active customer - blind		Document
Mobile	PICKSOACTIVECREATE	Pick SO Active Create		Document
Mobile	PICKSOACTIVELIST	Pick SO by active customer - list		Document
Mobile	PICKSOALLBLIND	Pick SO all - blind		Document
Mobile	PICKSOALLCREATE	Pick SO All Create		Document
Mobile	PICKSOALLLIST	Pick SO all - list		Document
Mobile	PICKSOLOCALBLIND	Pick SO by local customer - blind		Document
Mobile	PICKSOLOCALCREATE	Pick SO Local Create		Document
Mobile	PICKSOLOCALLIST	Pick SO by local customer - list		Document
Mobile	PICKTOACTIVEBLIND	Pick TO by active location - blind		Document
Mobile	PICKTOACTIVECREATE	Pick TO Active Create		Document
Mobile	PICKTOACTIVELIST	Pick TO by active location - list		Document
Mobile	PICKTOALLBLIND	Pick TO all - blind		Document
Mobile	PICKTOALLCREATE	Pick TO All Create		Document
Mobile	PICKTOALLLIST	Pick TO all - list		Document
Mobile	PICKTOLOCALBLIND	Pick TO by local local - blind		Document
Mobile	PICKTOLOCALCREATE	Pick TO Local Create		Document

Mobile	PICKTOLOCALLIST	Pick TO by local locations - list	Document
Mobile	RECEIVEPOACTIVCREATE	Receive PO Active Create	Document
Mobile	RECEIVEPOACTIVEBLIND	Receive PO by active vendors - blind	Document
Mobile	RECEIVEPOACTIVELIST	Receive PO by active vendors - list	Document
Mobile	RECEIVEPOALLBLIND	Receive PO all - blind	Document
Mobile	RECEIVEPOALLCREATE	Receive PO All Create	Document
Mobile	RECEIVEPOALLLIST	Receive PO all - list	Document
Mobile	RECEIVEPOITEMBLIND	Receive PO by item selection - blind	Document
Mobile	RECEIVEPOITEMLIST	Receive PO by item selection - list	Document
Mobile	RECEIVEPOLOCALBLIND	Receive PO by local vendor - blind	Document
Mobile	RECEIVEPOLOCALCREATE	Receive PO Local Create	Document
Mobile	RECEIVEPOLOCALLIST	Receive PO by local vendor - list	Document
Mobile	RECEIVESROACTIVEBLIN	Receive SRO by active customer - blind	Document
Mobile	RECEIVESROACTIVELIST	Receive SRO by active customer - list	Document
Mobile	RECEIVESROALLBLIND	Receive SRO all - blind	Document
Mobile	RECEIVESROALLLIST	Receive SRO all - list	Document
Mobile	RECEIVESROITEMBLIND	Receive SRO by item selection - blind	Document
Mobile	RECEIVESROITEMLIST	Receive SRO by item selection - list	Document
Mobile	RECEIVESROLOCALBLIND	Receive SRO by local customer - blind	Document
Mobile	RECEIVESROLOCALLIST	Receive SRO by local customer - list	Document
Mobile	RECEIVETOACTIVEBLIND	Receive TO by active locations - blind	Document
Mobile	RECEIVETOACTIVELIST	Receive TO by active locations - list	Document
Mobile	RECEIVETOALLBLIND	Receive TO all - blind	Document
Mobile	RECEIVETOALLLIST	Receive TO all - list	Document
Mobile	RECEIVETOITEMBLIND	Receive TO by item selection - blind	Document
Mobile	RECEIVETOITEMLIST	Receive TO by item selection - list	Document
Mobile	RECEIVETOLOCALBLIND	Receive TO by local location - blind	Document
Mobile	RECEIVETOLOCALLIST	Receive TO by local location - list	Document
Mobile	UNSENTDOCS	Unsent Documents	Document
Mobile	UNSENTWORKSHEETS	Unsent Worksheets	Worksheet
Mobile	UPDMASTER	Update master data	Function
Mobile	VENDORCARD	Vendor Card	Card
Mobile	VIEWDATARETRIEVED	View data retrieved	Function
Mobile	WHOAMI	Who am I	Function
Mobile	WORKLIST	Worksheet List	Worksheet
Mobile	WORKSHEET	Worksheet	Worksheet

Inventory Menus

List of menus


Inventory Menu List
Device Type Code Store No. Description Main Menu No. of Lines
Mobile LOOKUP MENU S0001 Lookup Menu 0
Mobile MAIN MENU S0001 Main Menu ✔ 0
Mobile NEG ADJ S0001 Negative Adjustment 0
Mobile PICKING S0001 Picking Menu 11
Mobile PICKING BLIND S0001 Picking Blind Menu 0
Mobile RECEIVING S0001 Receiving Menu 7
Mobile STOCK MAN... S0001 Stock Management 0
Mobile UTILITY S0001 Utility Menu 0
OK
Cancel
Help

Example of Menus and Menu Items. Following is a snapshot of content of the table Inventory Menu Lines (10012801)

Device Type	Menu Code	Store No.	Line No.	Description	Text	Line Type	Code	Bit	Status	Code Type	Mask Type	Mask Group	Mask No.	Location Code	Vendor No.
Mobile	LOOKUP MENU	S0001	10	Item Card		Code	ITEMCARD		Active	Card	Purchase		0		
Mobile	LOOKUP MENU	S0001	20	Item Card		Code	ITEMCARD		Inactive	Card	Purchase		0		
Mobile	LOOKUP MENU	S0001	30	Item Card		Code	ITEMCARD		Active	Card	Purchase		0		
Mobile	LOOKUP MENU	S0001	40	Customer Card		Code	CUSTOMERCARD		Active	Card	Purchase		0		
Mobile	LOOKUP MENU	S0001	50	Vendor Card		Code	VENDORCARD		Active	Card	Purchase		0		
Mobile	MAIN MENU	S0001	10	Utility Menu		Menu	UTILITY		Active		Purchase		0		
Mobile	MAIN MENU	S0001	15	Lookup Menu		Menu	LOOKUP MENU		Active		Purchase		0		
Mobile	MAIN MENU	S0001	20	Unsent Documents		Code	UNSENTDOCS		Active	Document	Purchase		0		
Mobile	MAIN MENU	S0001	30	Receiving Menu		Menu	RECEIVING		Active		Purchase		0		
Mobile	MAIN MENU	S0001	40	Picking Menu		Menu	PICKING		Active		Purchase		0		
Mobile	MAIN MENU	S0001	50	Stock Management		Menu	STOCK MANAGEMENT		Active		Purchase		0		
Mobile	MAIN MENU	S0001	60	Search for document		Code	DOCUMENTSEARCH		Active	Document	Purchase		0		
Mobile	MAIN MENU	S0001	70	Search for document blind		Code	DOCUMENTSEARCHBLIND		Active	Document	Purchase		0		
Mobile	NEG ADJ	S0001	10	Worksheet	Best Before Date	Code	WORKSHEET		Active	Worksheet	Negative		30000		
Mobile	NEG ADJ	S0001	20	Worksheet	Damaged By Customer	Code	WORKSHEET		Inactive	Worksheet	Negative		94000		
Mobile	NEG ADJ	S0001	30	Worksheet	Damaged By Staff	Code	WORKSHEET		Active	Worksheet	Negative		95000		
Mobile	PICKING	S0001	10	Pick SO all - list		Code	PICKSOALLLIST		Active	Document	Purchase		0		
Mobile	PICKING	S0001	20	Pick TO all - list		Code	PICKTOALLLIST		Active	Document	Purchase		0		

Mobile	PICKING	S0001	30	Pick PRO all - list		Code	PICKPROALLLIST	Active	Document	Purchase	0
Mobile	PICKING	S0001	40	Picking Blind Menu		Menu	PICKING BLIND	Active		Purchase	0
Mobile	PICKING BLIND	S0001	10	Pick SO by active customer - blind		Code	PICKSOACTIVEBLIND	Active	Document	Purchase	0
Mobile	PICKING BLIND	S0001	20	Pick SO by local customer - blind		Code	PICKSOLOCALBLIND	Active	Document	Purchase	0
Mobile	PICKING BLIND	S0001	30	Pick TO all - blind		Code	PICKTOALLBLIND	Active	Document	Purchase	0
Mobile	STOCK MANAGEMENT	S0001	10	Worksheet List	Positive Adjustment	Code	WORKLIST	Active	Worksheet	Positive	0
Mobile	STOCK MANAGEMENT	S0001	20	Positive Adjustment		Menu	NEG ADJ	Active		Purchase	0
Mobile	STOCK MANAGEMENT	S0001	30	Worksheet List	Counting Worksheets	Code	WORKLIST	Active	Worksheet	Counting	0
Mobile	UTILITY	S0001	10	Who am I		Code	WHOAMI	Active	Function	Purchase	0
Mobile	UTILITY	S0001	20	Update master data		Code	UPDMASTER	Inactive	Function	Purchase	0
Mobile	UTILITY	S0001	30	Full master data update		Code	FULLMASTER	Active	Function	Purchase	0
Mobile	UTILITY	S0001	40	View data retrieved		Code	VIEWDATARETRIEVED	Active	Function	Purchase	0

Appendix B

LS Retail provides a test program that can connect to the NAV web service using the same web methods as the Mobile Inventory application.

Pre-requirements

- Install the latest LS Retail demo company. The test programs should use the demo company since it already has all the objects needed.
- Make sure you have the Windows Mobile Device Center installed and that you are able to connect from your desktop to the mobile device via USB cable.
- The mobile device should be connected to a windows desktop machine via USB cable.

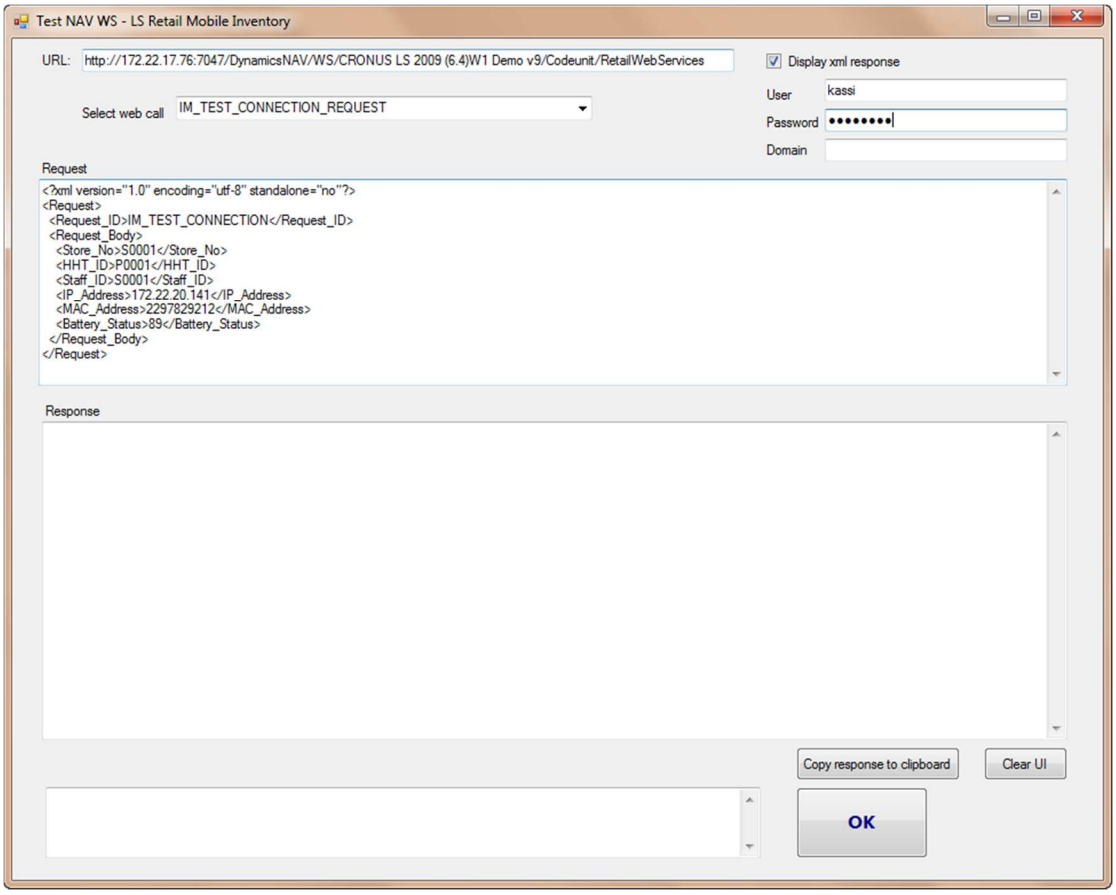
Connect to NAV web service from your desktop machine

Download the test tool from the partner portal

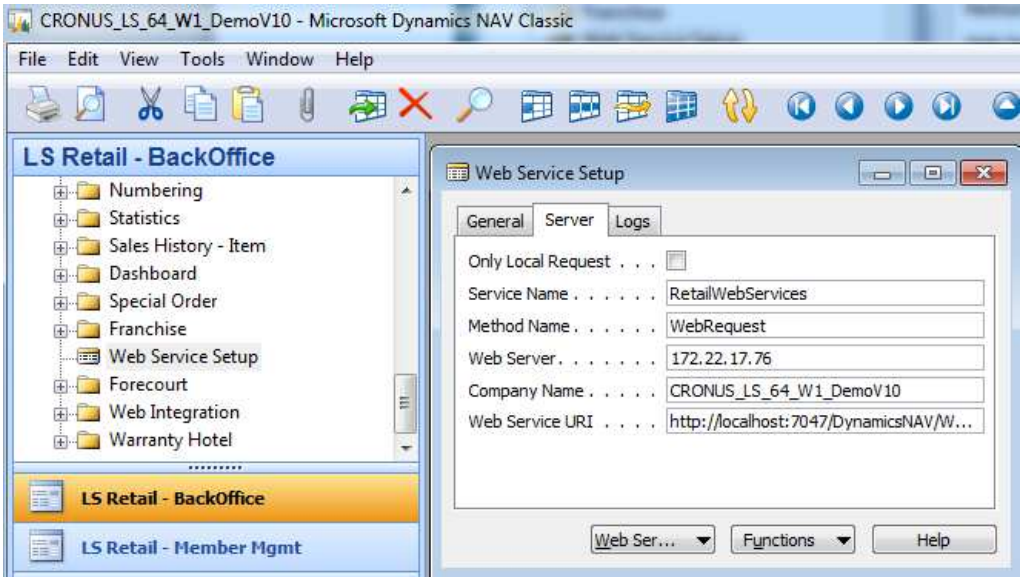
http://partners.lsretail.com/LSOmni/tabid/2475/language/en-US/Default.aspx

Unzip the TestNavWebService.zip and run the TestNavWs.exe from your desktop machine.

Enter the NAV user and password (this is a Windows user that connects to NAV web service), and the URL. This test application sends and retrieves data from NAV in XML format and is very helpful when debugging NAV web services.



- .Net 4.0 must be installed on your desktop.
- Enter the NAV user name and password.
  This password should **NOT** be encrypted using the MobileInventoryGenerator.exe.
- Enter the URL to the NAV web service. This URL should be the same as in the Web Service Setup.

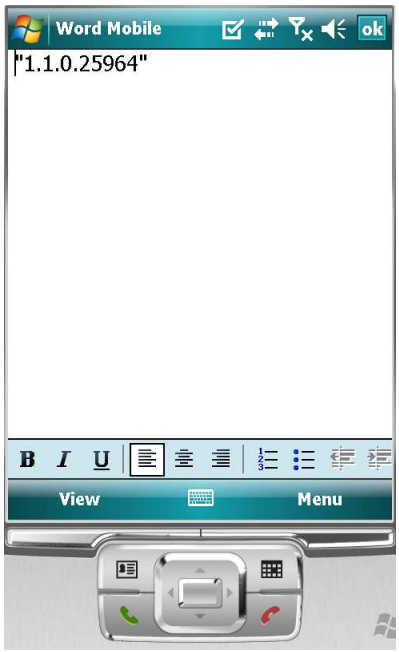


- This test application sends and retrieves data from NAV in XML format. You can change the values in the XML request body without breaking anything.
- Select a few different web calls and retrieve the data from NAV.
    - At startup the Mobile Inventory application uses the IM_TEST_CONNECTION, IM_GET_SETUP_DATA and IM_GET_WORK_DATA.
- The „Display xml response“ check box can be unchecked when you do not want to retrieve megabytes of data into the response textbox.

This test application can also be helpful when checking the data returned from NAV since it uses exactly the same web methods and xml data structure as the Mobile Inventory application.

Connect to INVService WCF web service from the mobile device

Open the internet browser. Type in the URL to the INVService. Ex. http://172.22.17.217/INVService/json.svc/version . If connection is successful a download window will appear. Save the response to a file which should show the version number.



Appendix C

The App.config file on the mobile device holds following configuration

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <appSettings>

    <add key="ApplicationTitle" value="LS Retail Mobile Inventory" />
    <add key="Terminal" value="P0001"/>
    <add key="Store" value="S0001"/>
    <!-- LogLevel (logs to logs\logfile.txt). level 1=trace, 2=errors only -
->
    <add key="LogLevel" value="1"/>

    <!-- IIS Inventory web service,  -->
    <add key="INVService.Url" value="http://localhost/INVService/Json.svc/"
/>
    <add key="INVService.TimeoutInSec" value="10"/>

    <!-- AllowDecimalInQty true/false, always allows the use of decimals in
qty (, or . depends on culture used) -->
    <add key="AllowDecimalInQty" value="false"/>

    <!-- Barcode.Mask.Prefix.StripCode  Comma delimited list of prefix-N
where only the first N digits are used to look up the item in local db
BarcodeTable. -->
    <!--   Ex. value="23-7,25-7"  then item lookup for barcode
"231234500110C" would be where the first 7 digits in BarcodeTable that match
"2312345" -->
    <!--   Used for weighted items, price in barcode etc where whole barcode
can not be used to look up the item -->
    <!--   Set value to empty string to disable this feature.  -->
    <add key="Barcode.Mask.Prefix.StripCode" value="23-7,25-7"/>
    <!-- ItemSearch.ShowItemNo, true shows the item no, otherwise only item
description is shown -->
    <add key="ItemSearch.ShowItemNo" value="false"/>

    <!-- Manually indicate whether the target device has a VGA screen -->
    <add key="ScreenResolution.VGA" value="true"/>

    <!-- CheckConnection.Timeout is the number of seconds before connection
timing out to NAV -->
    <add key="CheckConnection.Timeout" value="25"/>
    <!-- CheckConnectionSleep is the number of seconds between checking
connection to NAV. 0= no connection check done. -->
    <add key="CheckConnection.Sleep" value="30"/>
    <!-- CheckConnection.LogError (true/false). Connection errors can fill
up logtable. Set to false so error is not logged -->
    <add key="CheckConnection.LogError" value="true"/>
    <!-- DatabaseFile can also be "\Storage Card\Data\MobileInventory.sdf"
"Data\MobileInventory.sdf"-->
    <add key="DatabaseFile" value="Data\MobileInventory.db3" />

    <add key="ImageFolder" value="Img" />
    <!-- WorkerThread.Sleep = 0 turns off the workerthread. Value is number
of seconds the thread sleeps between work -->
    <add key="WorkerThread.Sleep" value="0" />
    <!-- WorkerThread.SendErrorLog = true to send errors to NAV -->
    <add key="WorkerThread.SendErrorLog" value="true" />
    <!-- WorkerThread.DBCleanupHour = 20  Hour of day when cleanup takes
place. Values from 1 to 23. 0 = no db cleanup -->
```

```
    <add key="WorkerThread.DBCleanupHour" value="20" />


    <!-- repositories used, do not modify -->
    <add key="Repository-File"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.dll" />
    <add key="RepositoryAgent"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Database.Reposit
oryAgent" />
    <add key="ItemRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Items.ItemReposi
tory" />
    <add key="CustomerRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Customers.Custom
erRepository" />
    <add key="LocationRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Locations.Locati
onRepository" />
    <add key="MaskRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Masks.MaskReposi
tory" />
    <add key="JournalLineRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Masks.JournalLin
eRepository" />
    <add key="CountingAreaRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Masks.CountingAr
eaRepository" />
    <add key="TerminalRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Terminals.Termin
alRepository" />
    <add key="MenuRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Menus.MenuReposi
tory" />
    <add key="DataImportRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.DataImport.DataI
mportRepository" />
    <add key="TransactionRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Transactions.Tra
nsactionRepository" />
    <add key="LogRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Logger.LogReposi
tory" />
    <add key="MaintenanceRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Database.Mainten
anceRepository" />
    <add key="DocumentRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.Documents.Docume
ntRepository" />
    <add key="DataSyncRepository"
value="LSRetail.Mobile.Inventory.Infrastructure.Data.Sqlite.DataSyncing.Data
SyncRepository" />

    <add key="BOConnection-File"
value="LSRetail.Mobile.Inventory.Infrastructure.BOConnection.INV.dll" />
    <add key="BOConnection"
value="LSRetail.Mobile.Inventory.Infrastructure.BOConnection.INV.NAVConnecti
on" />

    <!-- For connection to NAV web service use .NAV.dll -->
    <!--
    <add key="BOConnection-File"
value="LSRetail.Mobile.Inventory.Infrastructure.BOConnection.NAV.dll" />
    <add key="BOConnection"
value="LSRetail.Mobile.Inventory.Infrastructure.BOConnection.NAV.NAVConnecti
on" />
```

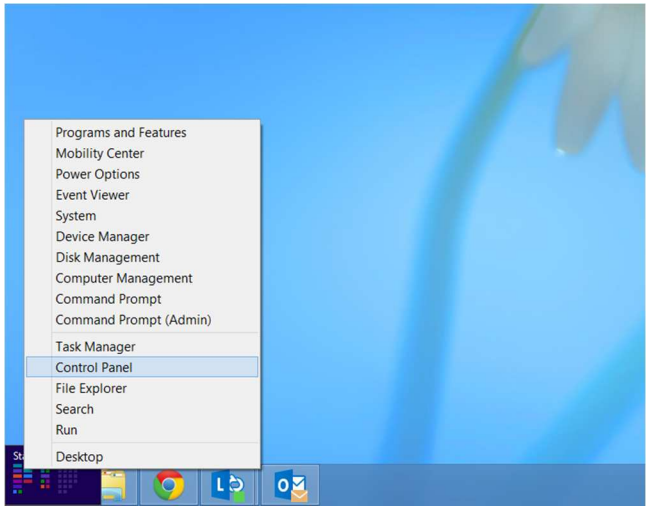
```
    -->

    <!-- Back office settigs, for connection to NAV web service -->
    <!--
    <add key="BOWebServiceURL"
value="http://localhost:7057/LSOmni/WS/CRONUSDemo14/Codeunit/RetailWebServic
es"/>

    <add key="BOWebServiceTimout" value="120"/>
    <add key="BOWebServiceUserName" value="NAV USER"/>
    <add key="BOWebServiceDomain" value=""/>
    <add key="BOWebServicePassword" value="NAV USER PASSWORD"/>
    <add key="BOWebServiceEncoding" value="utf-8"/>
    -->
    <!-- Encoding, usually utf-8 but chinese uses GB18030 -->
  </appSettings>
</configuration>
```

Appendix E

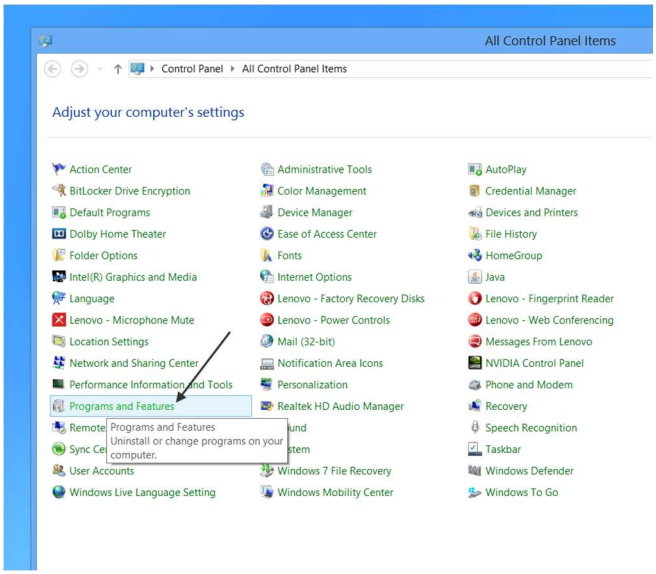
Installing IIS and WCF

The INVService web service requires IIS 7+ and Windows Communication Foundation (WCF). Windows 7/8/2008/2012 ships with IIS so it must be installed by turning on the Internet Information Services. Please follow these steps to turn on IIS 8 on Windows 8. The procedure is similar on other versions of Windows; a quick Google search will give multiple results on how to install IIS on your machine.

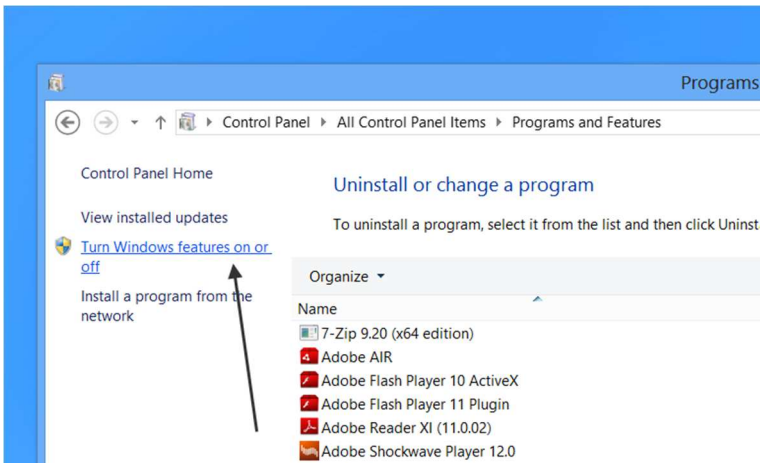
1. **On Windows 8:** Start by moving the mouse over to the bottom left corner, right click on Start and choose **Control Panel**:



2. Choose "**Programs and Features**":

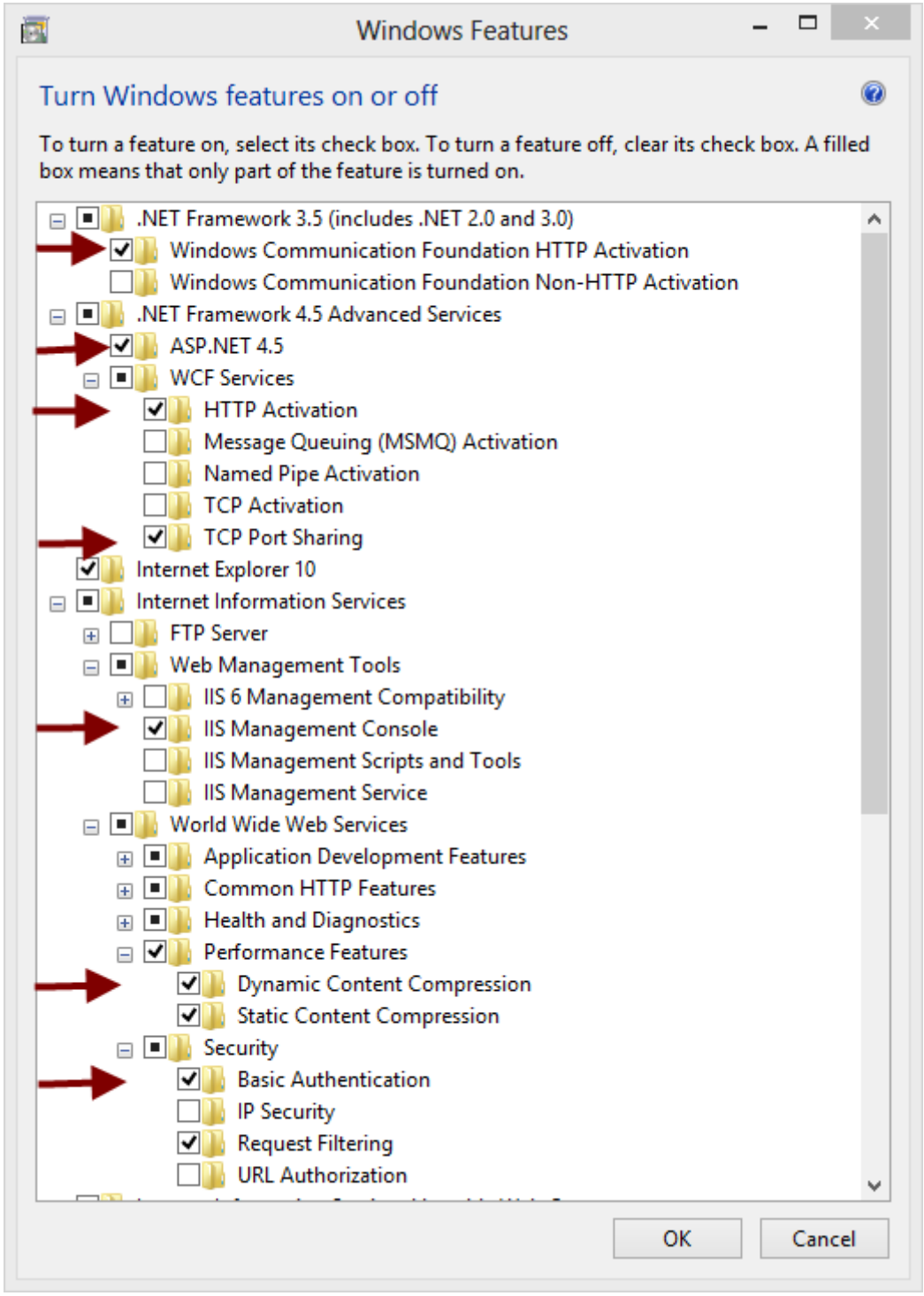


3. Choose "**Turn Windows features on or off**":



4. Select the checkboxes marked on the following picture :
+.NET framework 3.5 (includes .NET 2.0 and 3.0)
- Windows Communication Foundation HTTP Activation
+.Internet Information Services
- Web Management Tools
IIS Management Console
- World Wide Web Services
Dynamic Content Compression
Basic Authentication

Press **OK**.



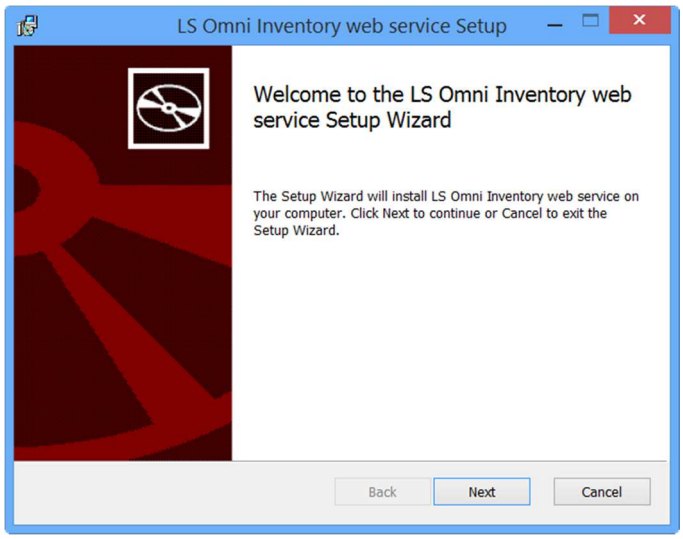
If you successfully completed step 4, then you should be ready to run the installation package to create the WCF web service.

Appendix F

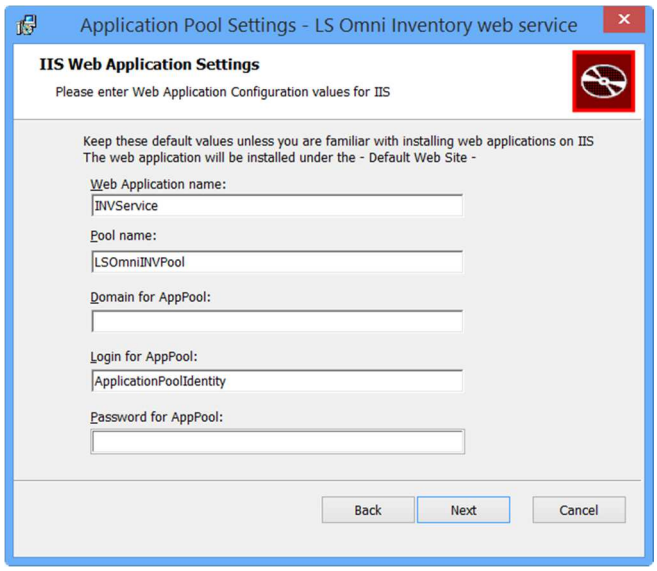
LS Omni Mobile INVService Installer package

Follow this step through guide when installing the INVService using Windows Installer package file. It can be uninstalled any time.

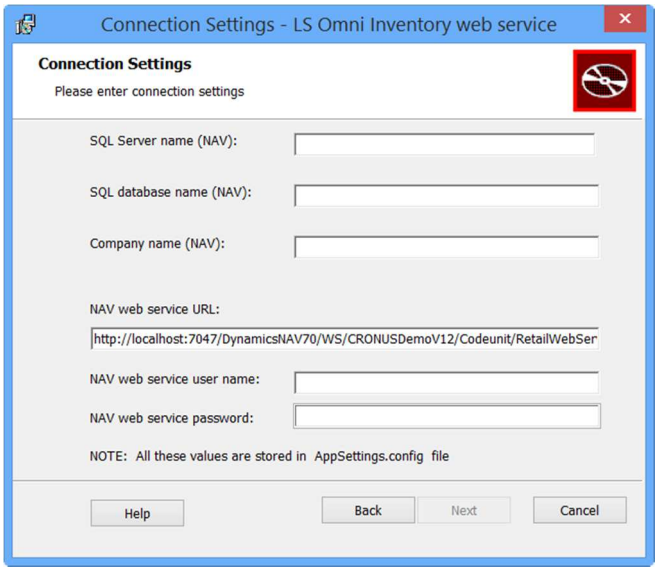
1. Double click the "LSOmni.Inventory.WcfServiceSetup.msi" file.
2. Choose **Next**:



3. Use the default values and click **Next:**



4. Fill out all input fields and click **Next.** If no domain or password is needed for AppPool you can leave those fields empty.



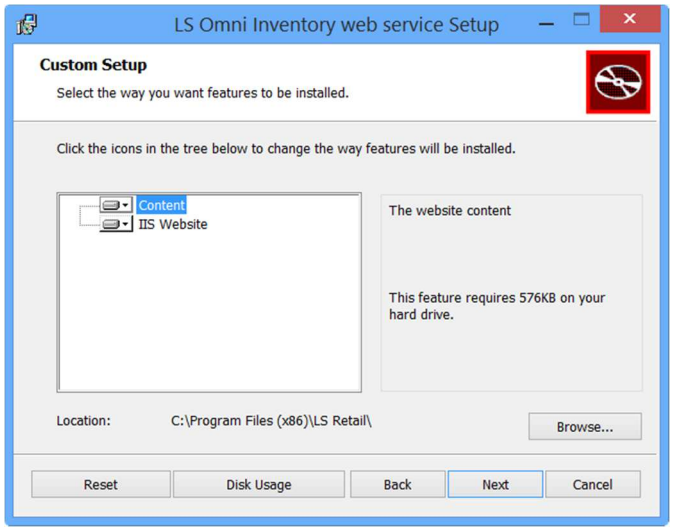
Nav web service URL:
`http://`***`NAVWEBSERVERNAME:PORTNO`***`(ie.7047)/`***`NAVSERVICEINSTANCENAME`***`(ie.DynamicsNAV70)/WS/`***`NAVCOMPANYNAME`***`(ie.CRONUS)/Codeunit/RetailWebServices` (Url to the NAV web service)

This URL should be the same as defined in NAV in the Web Service Setup, under Departments, LS Retail, Administration, Web Service.

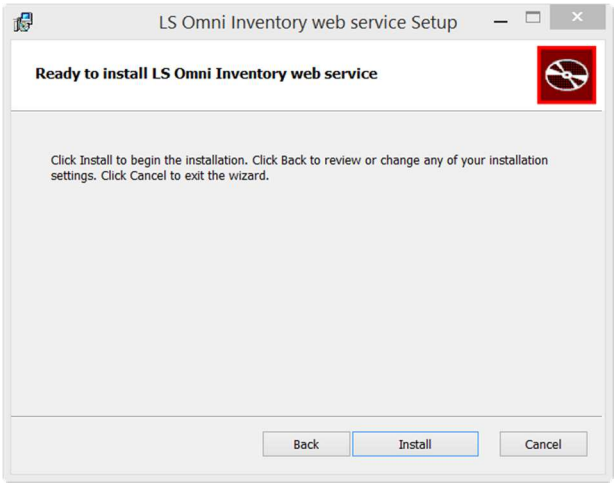
Nav web service user name: ***YourNavUser*** (is usually a windows user and is used to connect to the NAV web service, has role SUPER in NAV and is dbo or has equivalent rights in NAV database)**Nav web service password:** ***YourNavPassword*** (is the above YourNavUser password).

Test the connection to NAV web service with the URL, by pasting the URL into browser.

5. Select a location for the INVService files. The default folder is c:\Program Files (x86)\LS Retail\. Click **Next**

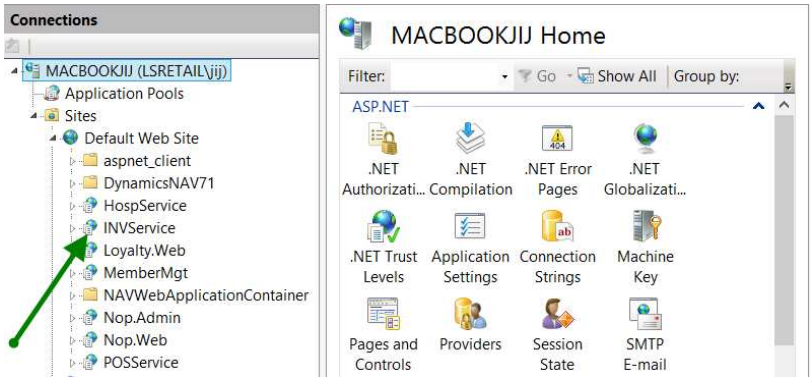


6. Click **Install**



7. Open **Internet Information Services (IIS Manager):**

   Validate that the INVService was created



8. Start the newly created Web Site. Validate it was successfully installed by checking the version from Chrome or Firefox: http://localhost/invservice/json.svc/version It returns the version "1.0.0.0" if INV web service is up and running.
9. Test the connection to NAV with this URL: http://localhost/invservice/json.svc/ping .

All errors are logged to \INVService\logs\logfile.txt file. Please check this file for detailed error messages.