8/1/2016

Vanguard Sales Manager

User Manual

Version 5.1

Philip Sheard
VANGUARD SOFTWARE

Contents

Prologue

Documents

Documentation for the Sales Manager is split into two volumes:

1. The Guide Book is a brief description of how to use the mobile app. It resembles the guide books that accompany most mobile devices, and its aim is similar: to provide the minimum amount of information necessary to accomplish the task.
2. The User Guide (this document) is more of a conventional manual. It aims to provide a comprehensive description of the whole system, and is aimed at system administrators rather than mobile users.

Style

There are few hyperlinks in this document. This is intentional, as they can cause a lot of clutter, and it only takes the reader a few seconds to copy and paste text into a search engine. Nor is any great use made of bold or italic text, and there is certainly no convention regarding their use.

Introduction

Overview

The Vanguard Sales System is a mobile field sales system, which runs on Android and iOS phones and tablets. It is a cloud-based client-server system, in which customer and product data is downloaded from a web server, and completed sales orders are uploaded back to the server. The data on the server is duplicated on the mobile device, allowing orders to be taken without the need for an internet connection. A connection is only required when uploading and downloading data.

Web Server

The system offers a choice of web server:

1. A company can host its own web server. Vanguard Software provides basic scripts for both the PHP/MySQL and ASP.Net/SQL Server environments, but companies should have at least some experience in the relevant technologies.
2. Vanguard Software provides a hosting service, which gives each subscriber his own virtual web server. This server is maintained using a web app, which will run in any modern browser. No advanced knowledge is required.
3. Individual users can opt to run the mobile app without a web server. This means that all their data has to be entered manually. Any features that rely on web queries, such as stock levels and customer history, will not work at all.

Costs Involved

Two different costs are involved when using the Vanguard Sales Manager:

1. In-app purchases are transactions made in the mobile app, and are processed through Google Play or iTunes. Vanguard Software has no control over these transactions. It cannot cancel them, or even confirm that they have taken place.
2. Licenses and subscriptions are transactions made directly with Vanguard Software, and are processed by PayPal. They can be carried out on the company web site, or via a PayPal invoice.

In-app purchases unlock various features of the app, such as the ability to send invoices by email. Each feature only has to be purchased once, **per Google Play or iTunes account**. Once features have been purchased on one device, they can be used on any other device, by tapping on the Restore button on the Store screen. The only proviso is that **the same account must be used on both occasions**. This applies to both Android and Apple devices.

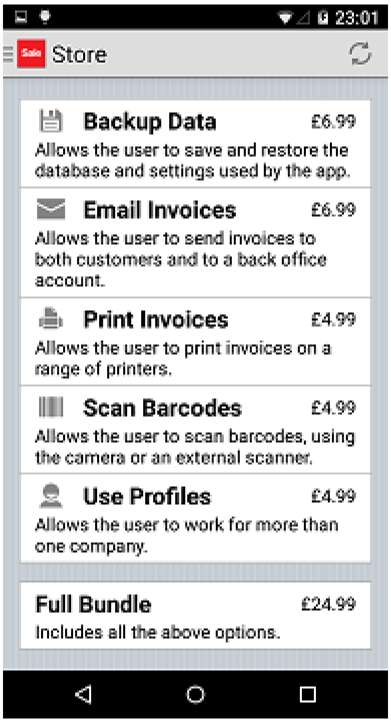
If a user wishes to sync his device with a self-hosted web server, he must purchase a license from Vanguard Software. In return he receives a registration key, which is unique to that device. The device id used to generate the key is shown on the About screen. A license is always for one specific device, and cannot be transferred to another. A license is valid for the lifetime of that device.

The Vanguard Software hosting service provides subscribers with their own virtual web servers, without the need to maintain real ones. These virtual web servers are maintained using a web app that can be run on any browser, and no prior knowledge is required. Subscribers pay a monthly or annual subscription. The cost depends on the number of users, and is on a sliding scale. The fee covers the use of a virtual web server, and the web app used to maintain it. Subscriptions have no minimum period, and can be cancelled at any time.

In-app Purchases

Store Screen

The screenshot on the right shows the Store screen for the Android app. The button in the action bar can be used to restore any purchases previously made for the current Google Play account, as discussed above. The items in the list below it are the various options that can be purchased.



Backup Data £6.99
Allows the user to save and restore the database and settings used by the app.

Email Invoices £6.99
Allows the user to send invoices to both customers and to a back office account.

Print Invoices £4.99
Allows the user to print invoices on a range of printers.

Scan Barcodes £4.99
Allows the user to scan barcodes, using the camera or an external scanner.

Use Profiles £4.99
Allows the user to work for more than one company.

Full Bundle £24.99
Includes all the above options.

Backup Data

This option allows the user to back up his database to a Dropbox account, and a free 2GB account is perfectly adequate for this purpose. Vanguard Software **strongly recommends** that all users who run the app without a web server purchase this option, and use it to back up their databases on a regular basis.

Email Invoices

This option allows the user to send a copy of each finished invoice to the customer, and another copy back to his company. This is a very useful and popular option. A Gmail account is required, and for security reasons we recommend that the user creates a dedicated one for use with this app.

Print Invoices

This option allows the user to print a copy of each finished invoice. Unfortunately this otherwise highly desirable feature has limitations which make it unsuitable for most mobile sales. The app will only print on A4 or US Letter sized stationery, and the printer **must have an internet connection**. It is not enough just to carry a mobile phone and a printer. The user has to carry a Wi-Fi router as well.

In practice this option is only suitable for scenarios such as trade shows and showrooms. It will also work with vans that have their own Wi-Fi routers and mains power supplies (these do exist, though they are rare). It works well with HP ePrinters, and it works with Google Print too, though some users have trouble getting this to work.

Scan Barcodes

This option supports Bluetooth scanners which support the HID protocol. This includes most of the scanners currently on sale, which are usually described as being Apple compatible. The Android app still supports camera scanning, but this is a bit of a party trick, and this feature may be withdrawn in future releases.

Use Profiles

This option allows the app to be used for more than one company. The database for each company is stored separately, allowing them to be hot-swapped. Since most of the user preferences are also stored in the database, this allows the company to be swapped immediately.

Full Bundle

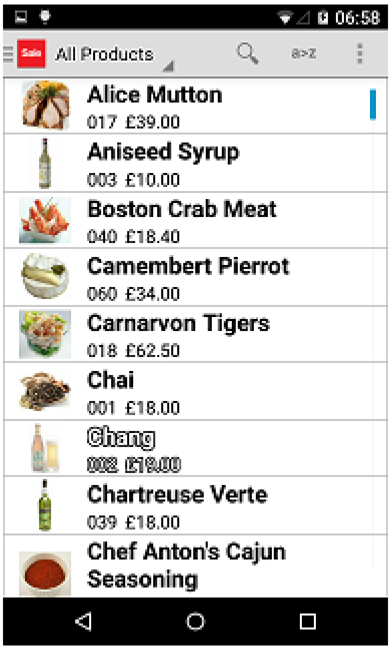
This is just a cheaper way to purchase all of the above options.

Product Images

Usage

Every product can be assigned an image. These images are stored on the mobile device, and are displayed individually, in grids, and as thumbnails. Each image is stored as a JPEG file, with a name matching the product id, and an extension of ".jpg". For a product to have an image, its id must be a valid Android file name.

It is a mistake to use large product images. Not only will they take longer to download, but the app has to reduce their size before it can process them. Android only allows apps to use a tiny amount of memory for images. 100KB is perfectly adequate, and most of the sample images are much smaller than that.



Alice Mutton
017 £39.00

Aniseed Syrup
003 £10.00

Boston Crab Meat
040 £18.40

Camembert Pierrot
060 £34.00

Carnarvon Tigers
018 £62.50

Chai
001 £18.00

Chang
002 £19.00

Chartreuse Verte
039 £18.00

Chef Anton's Cajun Seasoning

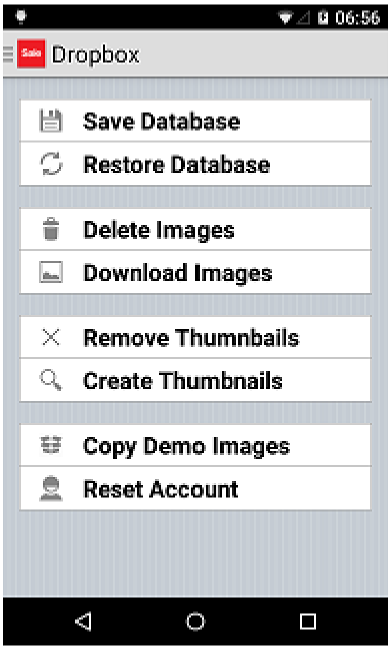
Dropbox Account

The product images are installed by storing them in a zip file, and downloading it from a Dropbox account. A free 2GB Dropbox account should be fine, depending on the number and size of the images. The account is linked to the app when the Dropbox screen is loaded for the first time. When the user agrees to allow the app to use the account, it creates an app folder called "Apps/Sales Manager". All the files that the app uses are contained in this folder. The one for the product images is called "images.zip".

Installation

To upload the product images:

1. Convert the images to JPEG format if necessary. Resize and crop them to taste, and rename them as appropriate. Ideally the images should be roughly square.
2. Place all the images in a zip file named "images.zip". Do not try to archive the folder containing the images, as this will not work.
3. Create a Dropbox account and link it to the app, as described above.
4. In a web browser, open the newly created app folder.
5. Upload the zip file.



To download the product images:

1. Download the Products table, if necessary. The product images are stored in the Products table. This means that they have to be generated every time that new product data is downloaded.
2. Go to the Dropbox screen.
3. Tap on Download Images, followed by Create thumbnails.

Platform Differences

Guide Book

The basic use of the mobile app is covered in detail in the Guide Book. This is a separate document aimed mainly at field users, which can be downloaded from the Vanguard Software web site. It describes the Android app specifically, but the user interface for the iOS app is very similar. It covers the use of the app on both phones and on tablets.

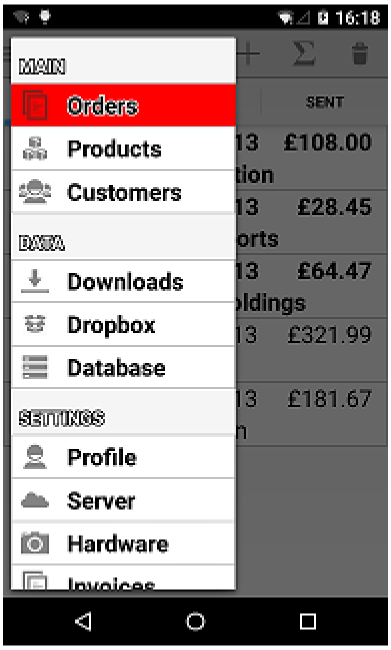
Platform Dependency

Most of the differences between the two apps are due to differences in the underlying platforms. Android and iOS have different design criteria, and each app is developed as a native app for its own platform. No attempt is made to copy features directly from one app to the other, without changing them to match the design standards of the new platform.

Some features are only available on one of the two platforms (usually Android). For instance Android devices have SD slots, while iOS ones do not. When this happens, the decision whether to use the feature at all is taken on merit. Some minor features can be omitted altogether. Others are so important that they have to be included, albeit on one platform only.

Navigation Drawer

The Sales Manager mobile app has a large number of screens, and navigating between them is particularly important. The traditional hierarchical menu structure offered by both Android and iOS is slow and time consuming. Android offers another solution for larger apps, called the Navigation Drawer. This allows the developer to present a menu for the entire app, as a drop-down list. The top part of the navigation drawer for the Android app is shown on the right.



The concept of the navigation drawer first appeared on iOS apps. But Android was quick to adopt it and incorporate it into their operating system, while Apple was not. For the Vanguard Sales Manager it is a must-have, killer feature. When the user launches the Android app for the first time, he knows that it may have a navigation drawer. So he taps the home button, and voilà! Unfortunately no such paradigm exists on the iOS platform, and the iOS app is stuck with an old-fashioned, hierarchical menu system.

Future of the iOS App

Vanguard Software is committed to maintaining the iOS app, and updating it in line with the Android version, far as possible. We have a design for a navigation drawer for the iOS platform, but the iOS app also lacks other important features, such as split screen mode and user-defined fields

Using both Platforms

It is possible to use a mixture of Android and iOS devices, but there are a number of caveats:

1. Separate payments must be made to Google Play and iTunes. Payments made on one platform cannot be transferred to the other.
2. It may not be possible to transfer an SQLite database from one platform to the other, due to their running different versions of SQLite. It used to be possible before they diverged, but now it is probably only possible one way round.
3. On a brighter note, all versions of the web server support both platforms equally well. The transactions are the same in both cases.

Upgrades

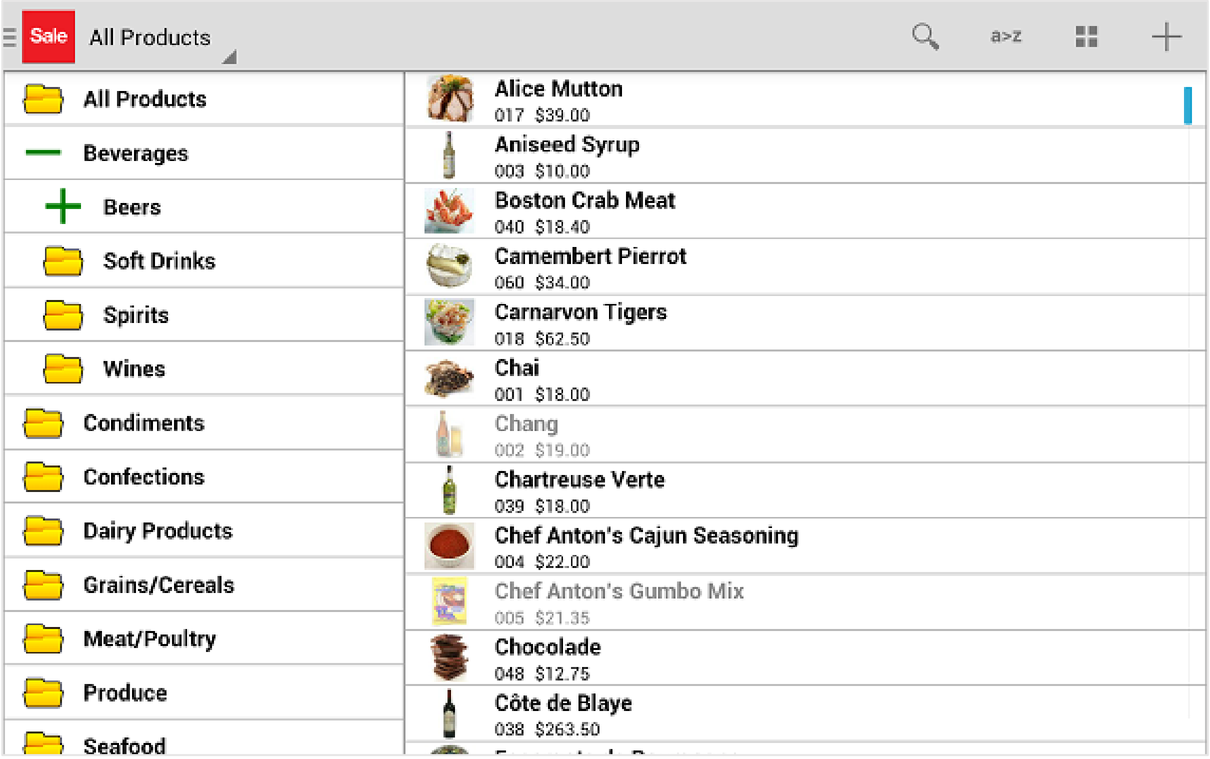
Upgrading the mobile app on either or both platforms should not cause any problems. Whenever a transaction changes, new PHP and ASP.Net scripts are created. The full name of the transaction includes the version number, e.g. products2.php. Older versions of the app continue to use older versions of the transaction, while newer versions use the new one.

1. The hosting service supports all versions of both mobile apps. The Vanguard Software server contains all versions of every PHP transaction.
2. The scripts downloaded from the Vanguard Software web site only include the latest version of each transaction, and this is the only version that we can fully support.
3. The current version of the Android app can also be downloaded from the Vanguard Software web site, as an APK file. This can be **sideloaded** to an Android device that is connected to a PC via a USB cable. There are plenty of descriptions of this process on the internet.
4. The Google Play and App Store servers always contain the latest version of the Android and iOS apps.

It is possible to fully lock down a self-hosted web server syncing with Android devices only, as follows:

1. Copy the server side scripts, and do not change them.
2. Download the APK file at the same time.
3. Turn off automatic upgrades on every Android device.
4. Sideload the APK file onto every Android device.

Split Screen



Alice Mutton	017	$39.00
Aniseed Syrup	003	$10.00
Boston Crab Meat	040	$18.40
Camembert Pierrot	060	$34.00
Carnarvon Tigers	018	$62.50
Chai	001	$18.00
Chang	002	$19.00
Chartreuse Verte	039	$18.00
Chef Anton's Cajun Seasoning	004	$22.00
Chef Anton's Gumbo Mix	005	$21.35
Chocolade	048	$12.75
Côte de Blaye	038	$263.50

Although it is not quite as important as the navigation drawer, split screen mode is another feature that is only available on Android. iOS offers some limited support for split screen mode, but not in a form that can be used by the mobile app. Split screen is only available on Android tablets, in horizontal mode.

Location and Maps

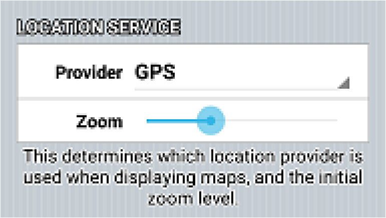
Location Services

The mobile app displays two types of map:

1. A map showing the positions of all the customers near the current location.
2. A map showing the route from the current location to a given customer.

Both maps require the current location, which can be determined either by GPS or from cell tower and Wi-Fi signals. GPS is slow, does not always work, and not all mobile devices have it. Network location is faster and nearly always works, but it is less accurate. In most cases though, it is sufficiently accurate.

The app allows the user to choose which location service to use. In the hardware settings, he can choose either GPS or network as the provider. If he finds that a map is not appearing, or is taking too long to load, he should switch to the network provider. The Zoom setting determines the initial size of the customer map.



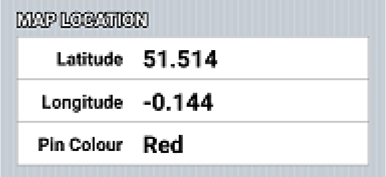
LOCATION SERVICE

Provider GPS

Zoom

This determines which location provider is used when displaying maps, and the initial zoom level.

The app stores the location of each customer in its customer record. If the location of a customer is not set, the customer does not appear on the customer map. The user can ask the app to determine the location of a customer by tapping the Locate button on the Edit Customer screen.



MAP LOCATION

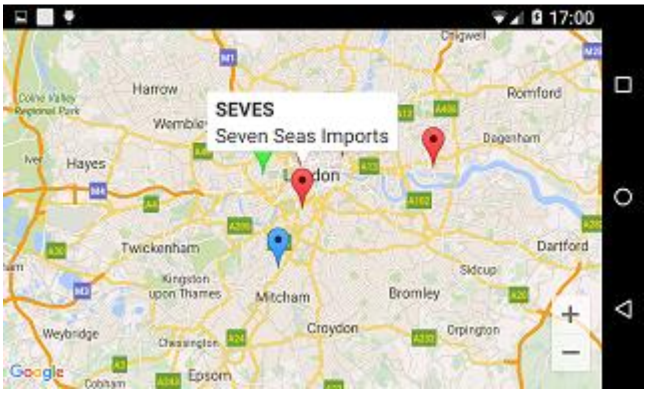
Latitude 51.514

Longitude -0.144

Pin Colour Red

The Locate function does **not** use either service provider, because the user might not be at the customer location. Instead it passes the address of the customer to the Google Maps server, and asks it to determine the location. This process is known as geocoding. Subscribers to the hosting service can ask Vanguard Software to geocode all their customers. There is no charge for this service.

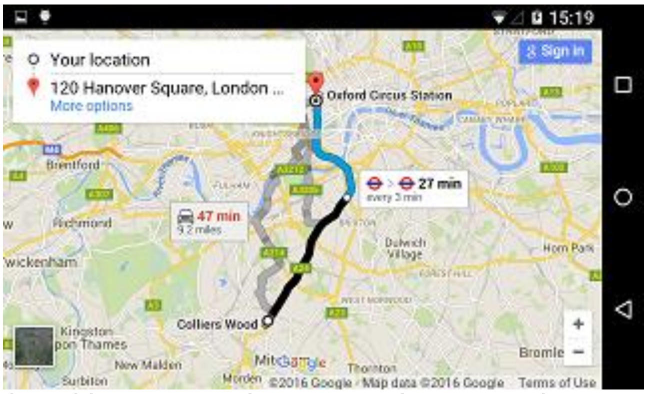
Customer Map



The Customer Map is displayed by tapping the Map icon in the action bar of the Customers screen. Above is an example for the London area, produced using the sample data. Tapping one of the pins pops up an annotation box, as shown above. This contains the id and name of the customer, and tapping it displays the customer record.

The light blue pin shows the current location. The pins for the customers are red, green and purple, as defined in the customer record.

Route Map



The Route Map is displayed by tapping the Contact button in the action bar on the Customer screen, and selecting Route Map from the drop-down list. The route map is produced by the standard Google Maps web site, and displayed in a web page. This allows the user to change the parameters used to determine the route (transport type, time etc.).

Using both Maps

It is possible to use both maps in tandem, as follows:

1. Go to the Customers screen.
2. Select the Map option.
3. On the customer map, identify a nearby customer to visit (perhaps by colour).
4. Tap the pin, and check the annotation.
5. If you have the right customer, tap the annotation, to go to the customer record.
6. Tap the Contact button in the action bar. The drop-down list contains various options for contacting the customer by phone, email etc.
7. If a visit is appropriate, select the Route Map option.
8. Use the route map to navigate to the customer.
9. Tap the back button, to return to the customer record.
10. If the customer is ready to place an order, tap the Order button in the action bar of the Customer screen.
11. Once the order has been taken and sent, tap the back button to return to the Customer screen.
12. Tap the back button again, to return to the customer map.
13. Repeat from step 3.

Geotagging

Orders uploaded to the web server can be tagged with the current location of the mobile user. This is enabled by turning on the geotagging option in the Invoices settings of the mobile app. Each order header in the Orders table has fields for latitude and longitude. If geotagging was enabled for the order, these will contain the location of the user when the order was uploaded.

User Interface

Orientation

One of the design objectives of the mobile app is that all screens should be viewable in both horizontal mode and vertical mode. This is especially useful for the iOS app, where fields do not adjust automatically to accommodate their content. Note that rotating a screen actually restarts its activity – something that you might not want to do when downloading a large table.

Swiping

A number of screens use horizontal swiping (both left-to-right and right-to-left) to move between the records in a list. It is available on most of the longer lists, such as customers and products. This is why edge swiping cannot be used to open the navigation drawer, as on most apps. The only way it open the navigation drawer is to tap on the home icon.

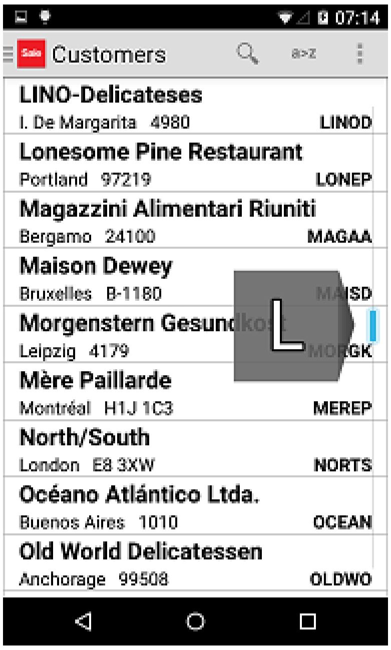
Fast Scroll

Fast scroll is a mechanism that allows the user to scroll through a list of data much faster than usual. All the larger lists in the mobile app, such as products and customers, support this feature. To use it, tap and hold the tab on the scroll bar.

If a large letter appears, as in the screenshot on the right, then the list supports fast scrolling. To use it, move the tab on the scroll bar, while keeping it held down. The list will scroll much faster than it usually does. The large letter is the first character in the name or id of the item at the top of the screen (depending on how the list is sorted).

Overflow Menu

When an action bar has too many options to be displayed on the screen, some of them are moved to an overflow menu. When this occurs, an overflow icon appears in the right had corner of the action bar, as shown in the screenshot on the right. When this icon is tapped, the remaining options are displayed as a drop-down list, as shown below.



LINO-Delicateses
I. De Margarita 4980 LINOD

Lonesome Pine Restaurant
Portland 97219 LONEP

Magazzini Alimentari Riuniti
Bergamo 24100 MAGAA

Maison Dewey
Bruxelles B-1180 MAISD

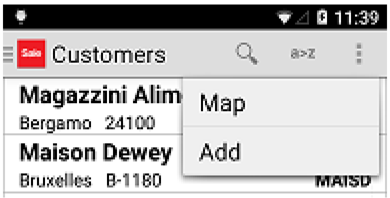
Morgenstern Gesundkost
Leipzig 4179 MORGK

Mère Paillarde
Montréal H1J 1C3 MEREP

North/South
London E8 3XW NORTS

Océano Atlántico Ltda.
Buenos Aires 1010 OCEAN

Old World Delicatessen
Anchorage 99508 OLDWO

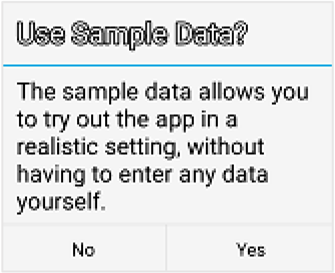


Magazzini Alim
Bergamo 24100

Maison Dewey
Bruxelles B-1180 MAISD

This is a standard feature of Android, but it is worth mentioning here because not all users are aware of it, and screens in the mobile app have more options than most. So if the text refers to say a map button on the Customers screen, and there does not appear to be one, the user should check the overflow menu. To see if it has been moved there.

Sample Data



Use Sample Data?

The sample data allows you to try out the app in a realistic setting, without having to enter any data yourself.

For this reason, when the mobile app is launched for the first time, it invites the user to download some sample data. If the user is trying out the app for the first time, this is recommended. Using the mobile app without any data at all can be confusing, and entering a significant amount of data by hand would be very tiresome.

The user can remove the sample data as follows:

1. Launch the Android Settings app.
2. Go to the Apps page.
3. Select the Sales app.
4. Tap on CLEAR DATA.
5. Launch the Sales app.
6. When asked whether you want to use the sample data, reply NO.

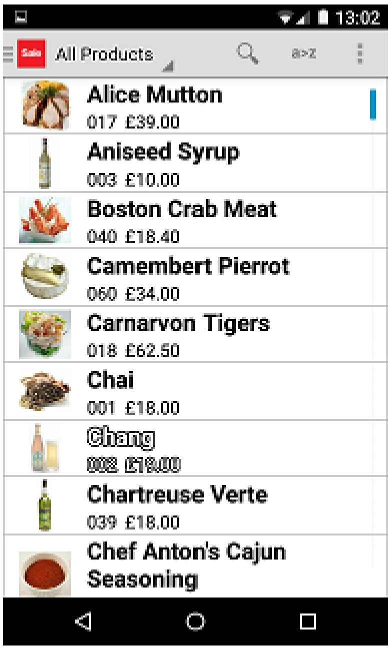
The user can install most of the sample data again, by downloading it from the demo company. To do this:

1. Go to the server settings.
2. Select demo company for the server type.
3. Go to the Downloads page.
4. Download each of the tables in turn.

Large Lists

Scalability

One of the features that distinguishes the Sales Manager from its competitors is its ability to handle large amounts of data. It can easily cope with 50,000 customers or products, with almost no loss of performance. To achieve this, the mobile app is heavily optimised, and provides a rich set of features to search such large datasets.



Alice Mutton
017 £39.00
Aniseed Syrup
003 £10.00
Boston Crab Meat
040 £18.40
Camembert Pierrot
060 £34.00
Carnarvon Tigers
018 £62.50
Chai
001 £18.00
Chang
002 £19.00
Chartreuse Verte
039 £18.00
Chef Anton's Cajun Seasoning

The lists involved are:

1. The customer list.
2. The product list.
3. The list of lines that can be added to an order.

The main difference between the customer and product lists is that the latter can be divided into categories. The lines behave in a similar way, but have additional properties. This chapter will focus primarily on the product list, but the others behave in a similar manner.

The name of the category is used as the title of the screen, and tapping on it displays a list of categories to choose from. This is a standard Android paradigm.

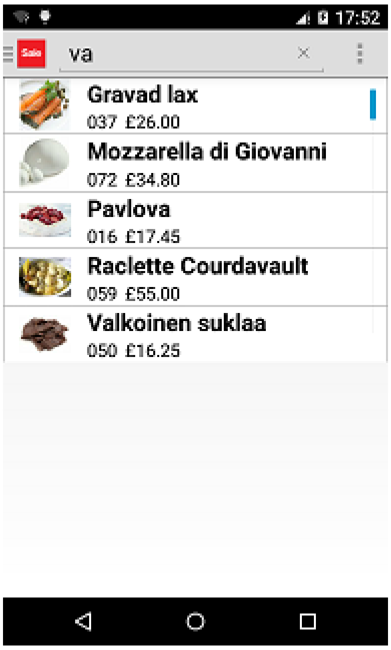
Sort Order

The list can be sorted either by id or by name. When a list is opened, it is always sorted by name, but this can be changed by tapping on the Sort button in the action bar. Changing the sort order also changes the field that gets searched. If the list is ordered by name, that is the field that gets searched, and vice versa.

Searching

To search a list:

1. Tap the Search icon (magnifying glass) in the action bar. The action bar is filled with an input field, and the input panel (soft keyboard) pops up.
2. Enter the target text to search for, and tap the back button, to dismiss the input panel. The list will now be restricted to the items whose names (or ids) contain the target text. If a category has been selected, only the items in that category will be displayed. An example is shown in the screenshot on the right.
3. Tap the back button, to end the search and return the list to its normal state.



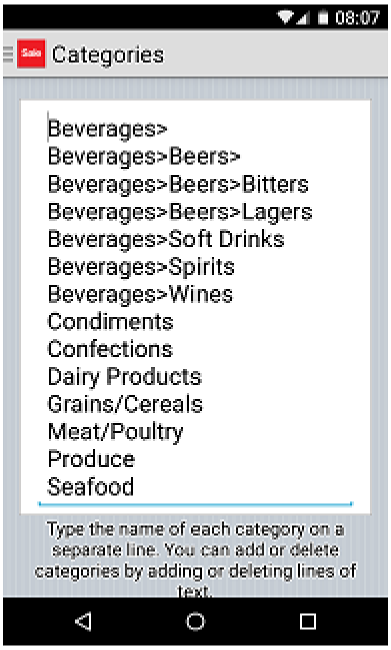
Gravad lax
037 £26.00
Mozzarella di Giovanni
072 £34.80
Pavlova
016 £17.45
Raclette Courdavault
059 £55.00
Valkoinen suklaa
050 £16.25

Note that the search is case insensitive – upper and lower case characters are treated in exactly the same way. Note also that only the key field is searched. It is not possible to search for text in any other field. This is done by design, in order to speed up the search process.

Categories

Structure

The product categories have a simple breadcrumb structure, which is well illustrated by the categories in the sample data, as shown in the screenshot on the right.



Beverages>
Beverages>Beers>
Beverages>Beers>Bitters
Beverages>Beers>Lagers
Beverages>Soft Drinks
Beverages>Spirits
Beverages>Wines
Condiments
Confections
Dairy Products
Grains/Cereals
Meat/Poultry
Produce
Seafood

Type the name of each category on a separate line. You can add or delete categories by adding or deleting lines of text.

Rules

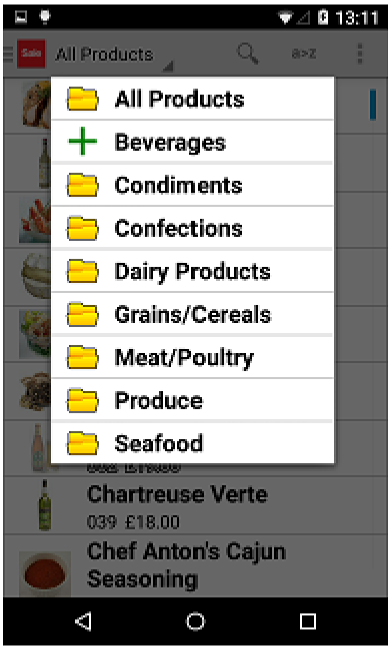
Categories must obey a few simple rules:

1. The “>” (greater than) character is used as a separator, so for example Wines is a subcategory of Beverages, and its full name is “Beverages>Wines”.
2. The full name of a category includes the names of all its parent categories. For instance Bitters is a subcategory of Beers, which in turn is a subcategory of Beverages. Its full name is “Beverages>Beers>Bitters”.
3. Categories can either contain products or other categories, but not both.
4. If a category contains other categories, its full name end in a separator, e.g. “Beverages>”.
5. A product can only belong to one category.

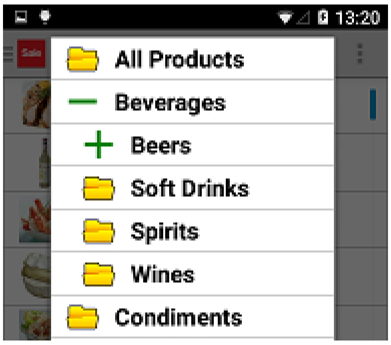
The last three rules are not intuitively obvious. They are included to make things easier, both for the user and the app.

Selection

Because categories can be nested, they are represented as a tree structure. The screenshot on the right shows a category being selected from the sample data. The icon for the Beverages category indicates that it contains other categories, but no products. Tapping this category expands it, as shown below.



Note that the Beverages category has expanded, and its icon has changed. Now tapping on it would cause it to contract again, hiding all its subcategories. Beers is another parent category, which shows that categories can be nested more than one level deep.



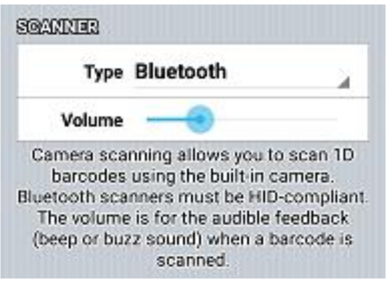
Barcode Scanning

Symbologies

The app supports just about any 1-D symbology, including UPC-A, UPC-E, EAN-8, EAN-13 and ISBN.

Camera Scanning

With a lot of clever programming, it is possible to scan 1-D barcodes on most modern smartphones. This is a spin-off from work done on the scanning of 2-D barcodes, which were specifically designed for this purpose. It is not as fast as using an external Bluetooth scanner, but with a little practice, it can be made to work reasonably well.



SCANNER

Type Bluetooth

Volume

Camera scanning allows you to scan 1D barcodes using the built in camera. Bluetooth scanners must be HID-compliant. The volume is for the audible feedback (beep or buzz sound) when a barcode is scanned.

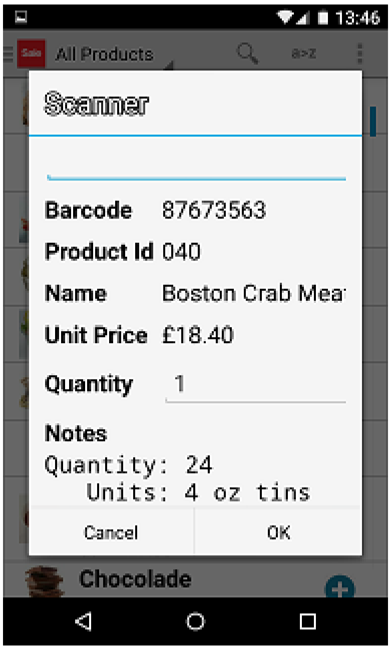
It is quite a party trick, and something that Vanguard Software has supported from the beginning. Android has a Google supported project called ZXing, and their library has always worked well. There was also a library for iOS called ZBar. It was less well supported and is now abandonware. It does not support the most recent versions of iOS.

Vanguard Software was forced to abandon camera scanning on the iOS platform when the ZBar project folded, and is now considering whether to drop it on Android as well. Camera scanning takes over the whole display, and integrating this into the overall dialog is a major design headache. It is probably much less of a problem when scanning 2-D barcodes, whose usage is quite different.

Bluetooth Scanners

The mobile app will work with any Bluetooth barcode scanner that supports the HID (Human Interface Device) protocol. This is the protocol used to connect devices such as keyboards and mice. Most modern scanners do so, and are often advertised as being “Apple compatible”. The scanner functions in much the same way an external keyboard. The advantage of this is that it does not require any additional drivers

In addition to running in HID mode, the scanner must be configured with no prefix, and a suffix of CR (carriage return). These are the characters added to the beginning and end of the scan data. In other words, the scanner just adds a carriage return character to the end of the data. This is the default for most scanners.



Scanner

Barcode 87673563

Product Id 040

Name Boston Crab Mea

Unit Price £18.40

Quantity 1

Notes

```
Quantity: 24
   Units: 4 oz tins
```

Scanning Dialog

The Android app uses a dialog for scanning barcodes, as shown on the right. At the top of the screen is the text box, which is currently empty. When a barcode is scanned, the data is inserted into the text box, character by character. When the app detects a carriage return character in the input data, it processes the barcode in the scan data, updates the display, and clears down the text box. If the OK button is tapped (after changing the quantity if necessary), the line is added to the order, and another barcode is scanned. Tapping Cancel ends the dialog.

Server Settings

Introduction

The Sales Manager is designed from the ground up as a cloud-based, client-server system. As mentioned previously, the mobile app can also be used as a stand-alone app, without a web server. However, all the information related to sales that the mobile app produces are web queries, and only orders that have been uploaded to the web server are taken into consideration. Any orders on the mobile device that have not been uploaded to the web server are ignored.

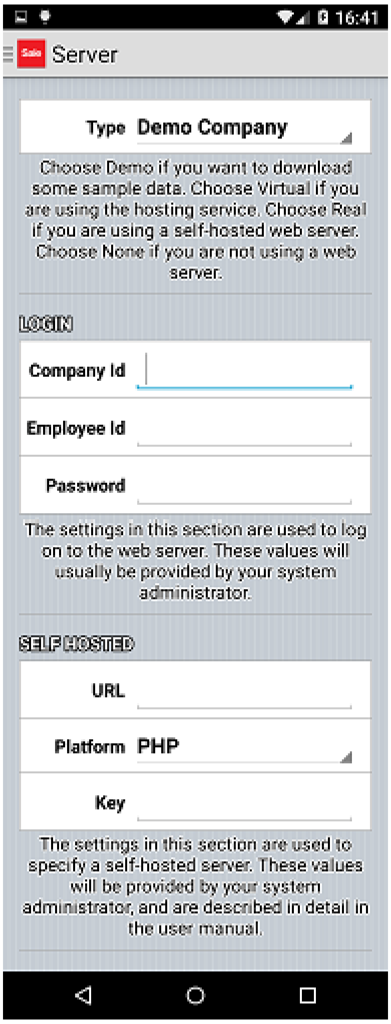
Server Type

There are four choices for the type of web server:

1. **Demo Company** allows the user to download the sample data. Warning: this will overwrite any data that the user might have entered manually.
2. **Hosting Service** allows the user to sync with a virtual web server on the Vanguard Software hosting service.
3. **Self Hosted** allows the user to sync with a real web server on the hosted by the user's own organisation.
4. **None** gives the best choice of menu options if the user is running without a web server. All server-related options are omitted.

Login

These are the details used to log in to the web server. Usually these will be supplied by a system administrator. For self-hosted web servers, the company id is not normally used.



Server

Type: Demo Company

Choose Demo if you want to download some sample data. Choose Virtual if you are using the hosting service. Choose Real if you are using a self-hosted web server. Choose None if you are not using a web server.

LOGIN

Company Id:

Employee Id:

Password:

The settings in this section are used to log on to the web server. These values will usually be provided by your system administrator.

SELF HOSTED

URL:

Platform: PHP

Key:

The settings in this section are used to specify a self-hosted server. These values will be provided by your system administrator, and are described in detail in the user manual.

Self Hosted

These are the details used define web server hosted by the user's own organisation. Usually these will be supplied by a system administrator.

1. These details are only required for self-hosted web servers: The URL identifies the folder containing the scripts for the web transactions. It should include the protocol, as the Android app supports HTTPS. The iOS version already requires HTTPS, as iOS no longer allows the use of the HTTP protocol for dedicated web servers.
2. The platform can be either PHP, ASP or ASP.Net. Please note that Vanguard Software no longer supplies scripts for classic ASP.
3. The registration key is supplied by Vanguard Software, and is valid for the current device only.

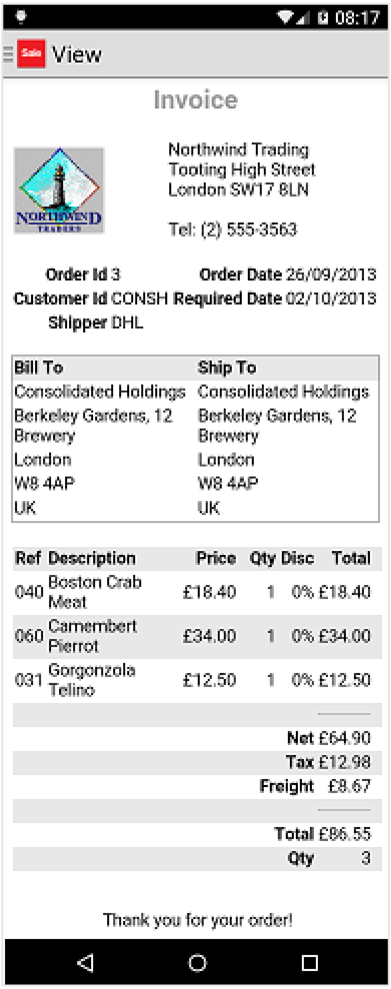
Invoices

Document

The mobile app generates invoices as HTML documents. This makes them easy to manage. Browsers and mail clients can display HTML documents directly, and services such as Google Print and HP ePrint can render them for printing on a variety of printers. HTML documents are also compact, making them easy to store and transmit.

HTML documents have a very simple format, which makes them easy to create. The only caveat is that the built-in HTML browsers in email clients such as Gmail tend to be very old-fashioned. Fancy CSS formatting is out, and it is better to use tables when laying out documents, even though it is generally frowned on nowadays.

A sample invoice is shown on the right, being viewed on a smartphone. This one has a custom HTML header, which includes a company logo. All the other options have been switched off.



Invoice

Northwind Trading
Tooting High Street
London SW17 8LN

Tel: (2) 555-3563

Order Id 3 **Order Date** 26/09/2013
Customer Id CONSH **Required Date** 02/10/2013
Shipper DHL

Bill To	Ship To
Consolidated Holdings	Consolidated Holdings
Berkeley Gardens, 12 Brewery	Berkeley Gardens, 12 Brewery
London	London
W8 4AP	W8 4AP
UK	UK

Ref	Description	Price	Qty	Disc	Total
040	Boston Crab Meat	£18.40	1	0%	£18.40
060	Camembert Pierrot	£34.00	1	0%	£34.00
031	Gorgonzola Telino	£12.50	1	0%	£12.50
				Net	£64.90
				Tax	£12.98
				Freight	£8.67
				Total	£86.55
				Qty	3

Thank you for your order!

Order Id

The first and most obvious value in the invoice settings is the order id. The app keeps track of the number of the last document produced (invoice, credit note or quotation), and increments it every time a new one is generated. In a multi-user environment, the system administrator assigns a different range of ids to each mobile device.

If two users accidentally use the same order id, the web server handles the situation quite easily, by storing both orders in its database. However, duplicate order ids would probably be disastrous for any accounting system that imports the orders stored in the web server. It is down to the system administrator to monitor the system for this, and to correct any errors that might arise as a result.

Header

The app can generate two types of header: plain text and HTML. Plain text headers are easy to use: the text in the header setting is simply copied into the centre of the page. An example of a plain text header for the above invoice would be:

```
Northwind Trading
Tooting High Street
London SW17 8LN

Tel: (2) 555-3563
```

For most users, plain text headers are the way to go. However, for the sake of completeness, below is the HTML header used to produce the invoice shown above:

```
<table width="100%">
<tr>
<td width="1%">
<img src="http://www.vanware.co.uk/images/northwind.gif" width="90" height="86">
</td>
<td>&nbsp;</td>
<td width="1%" nowrap="nowrap">
Northwind Trading<br>
Tooting High Street<br>
London SW17 8LN<br>
<br>
Tel: (2) 555-3563
</td>
<td>&nbsp;</td>
</tr>
</table>
```

The HTML has been tidied up, but is still pretty arcane. Creating code like this involves a lot of trial and error. Developers should also remember to tick the HTML format box below the header text.

Footer

The footer text is just like the header text, except that it appears at the bottom of the order instead of the top, and there is no HTML option. The text in the footer just gets centred on the page, and printed at the end of the document.

Options

There are five options for invoices:

1. **Tax Summary** prints a breakdown showing the amount of tax payed in each tax code.
2. **Custom** prints the values of any custom fields that might have been added to the order (see below).
3. **Notes** prints the contents of the order notes. Orders might contain either confidential information, or information that is an integral part of the order. Currently there is only one setting for the entire app.
4. **Attachment** adds a copy of the order in proprietary format (see below). This contains all the information in the order, in a machine readable form.
5. **Sort order** determines the order in which the detail lines are printed. This can be by id, by description or by insertion order (i.e. the order in which the lines are added).

Gmail Settings

The point to watch out for here is that the app sends emails without being directly authorised by the user. If Gmail thinks that an account is too important, it will not allow it to be used in this way. So users should avoid using any of their regular email accounts for this purpose.

How Gmail decides that an account is important is open to speculation. But this definitely will not happen if the account is created specifically to send emails, and only ever used for this purpose. This is also in line with our own security recommendations.

Formats

Field Types

The field types given below are named after the ones available in Microsoft Access, which are as follows:

Name	Contents
Boolean	A Boolean value.
Byte	A 1-byte integer.
Currency	A decimal value, with up to 28 digits.
Double	A 64-bit floating point value
Integer	A 2-byte integer.
Long	A 4-byte integer.
Memo	A string of arbitrary length.
Text	A short string with up to 255 characters.

Nullable Values

None of the fields in the database is nullable. The default values for all text strings is a zero-length string, and the default values for numeric values is zero.

Booleans

Booleans are stored as bytes, with the usual convention that zero is false, and any other value is true.

Percentages

Percentages are divided by 100, e.g. 20% is stored as 0.2. This is the standard method for representing percentages.

Notes

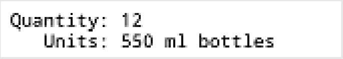
The notes fields in the customers, products and orders tables have a number of special properties:

1. They can contain any amount of text. Other fields have a maximum of 255 characters.
2. In CSV files, newline characters are represented by “\n”. This is true of other fields too, though it primarily applies to notes.
3. In the mobile app, notes are always displayed in a fixed font.
4. Strings that represent web addresses, email addresses, phone numbers and conventional addresses are displayed as clickable links.

The product notes in the sample data illustrate some of these properties. For instance, the notes for Aniseed Syrup (product id 003) are as follows:

```
Quantity: 12\n   Units: 550 ml bottles
```

When viewed in the mobile app, they look as shown on the right. The text breaks over two lines, and the two fields line up correctly.



```
Quantity: 12
   Units: 550 ml bottles
```

GPS Coordinates

Latitude and longitude are stored as integers, in millionths of a degree north and east. For instance, in the sample data, the company called Around the Horn is stored with a latitude of 51,521,006 and a longitude of -85,728, which translates to 51.521006° N, -0.085728° E. This was the original storage method used by Android.

Language

Encoding

The earliest computers used an 8-bit character encoding called ANSI. This was fine for West European languages, such as English or French, but did not have enough characters to represent languages such as Greek and Hebrew as well. The solution was a 16-bit character set called Unicode. Unicode text is usually stored in a compressed form called UTF-8. The Sales Manager supports UTF-8 throughout.

Locale

The mobile app formats dates, percentages, amounts of money etc. according to the locale that the user has chosen in his settings.

To change the system locale on Android:

1. Launch the Android Settings app.
2. Go to the Language & input page.
3. Tap the Language setting at the top of the screen.
4. Select a locale from the drop-down list.
5. Tap the back button.
6. Go to the Apps page.
7. Select the Sales app.
8. Tap FORCE STOP, followed by OK.

Internationalisation

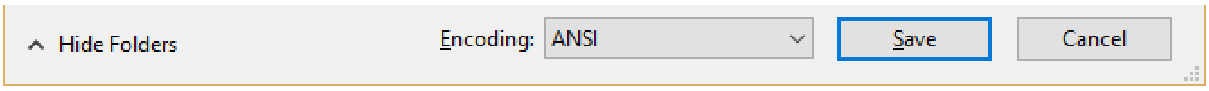
The Android app supports a number of foreign languages. Support for a given language is added by translating the text strings that the app uses, from English into the target language. A file containing the text strings can be downloaded from the Vanguard Software web site. Once translated, the file has to be built into the app. The app uses the current system locale to determine which language to use.

Two levels of language support are offered for the Android app:

1. All the text used by the app can be translated. This requires the translation of about 400 strings of text.
2. Only the text strings used in invoices can be translated. Clearly this is a much easier task.

For the iOS app, only invoices can be translated.

Windows


Encoding: ANSI

When running Windows, it can often be difficult to determine whether a given file is in ANSI or UTF-8 format. This is because Windows does a very good job of hiding it from us. It does that by adding an extra 3 bytes to the beginning of UTF-8 files, called the BOM. An easy way to tell though is to open the file in Notepad, and then select Save As from the File menu. The bottom of the Save As panel shows the current encoding, as shown above.

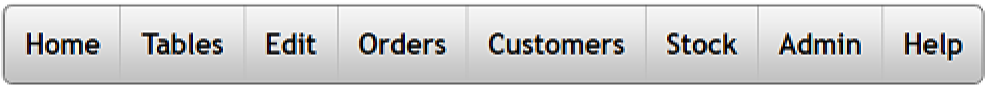
Hosting Service

Web App

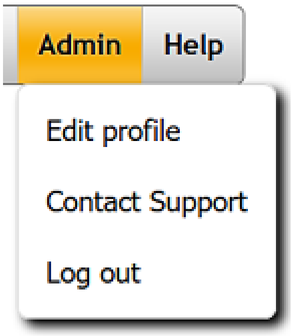
The Hosting Service allows a company to use its own virtual web server, without the hassle of hosting its own. The virtual web server is maintained by means of a web app, which will run in any browser. This chapter describes the use of that browser. The header appears at the top of every page, as shown above. The more modest footer is shown below.

© Vanguard Software 2000-2016

Menu Bar

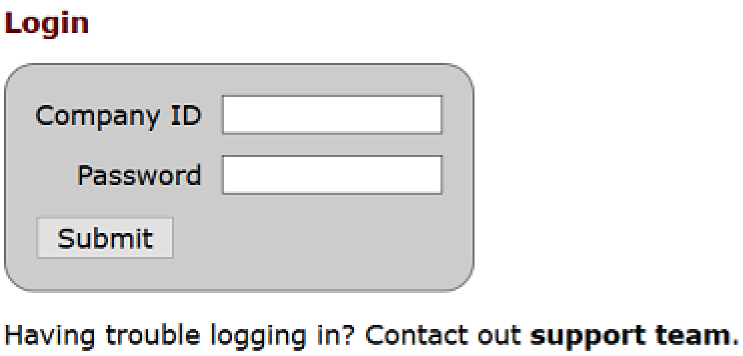


The menu bar is the main navigational tool for the web app, and contains standard drop-down menus, as shown on the right.



Login

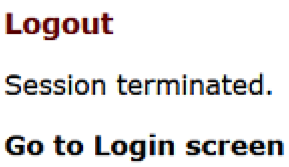
The login page appears when the web app is launched. It does not have a menu bar. A link to this can be found in the footer of the main Vanguard Software web site.



Having trouble logging in? Contact out **support team**.

Logout

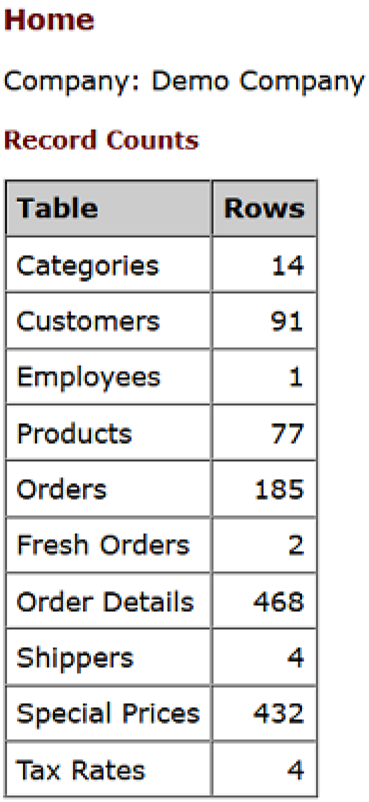
The Logout page appears when the user selects the Log out option from the Admin menu on the menu bar. It does not have a menu bar.



Session terminated.

Home Page

The Home page appears when the user has successfully logged in to the app. It shows the name of the company and the number of records in each table, as shown on the right.



Home

Company: Demo Company

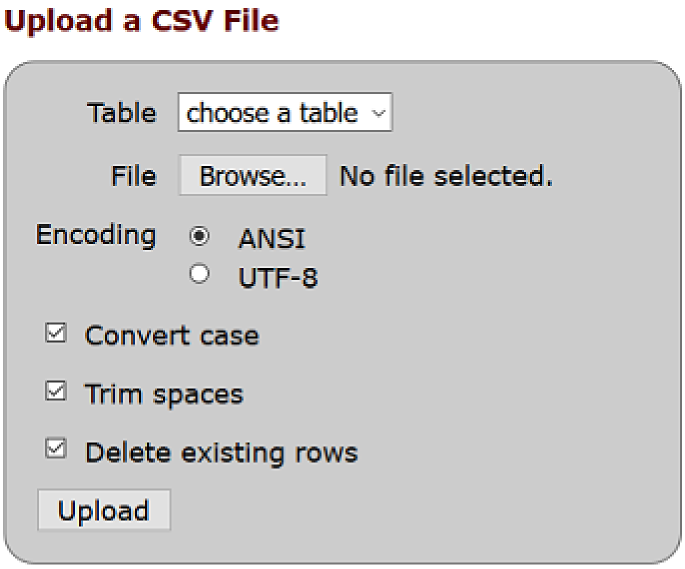
Record Counts

Table	Rows
Categories	14
Customers	91
Employees	1
Products	77
Orders	185
Fresh Orders	2
Order Details	468
Shippers	4
Special Prices	432
Tax Rates	4

Upload CSV File

The Upload a CSV File page appears when the user selects the Upload option from the Tables menu. You can use this page to upload any table in the database, though normally it would not be used to upload the orders or orderDetails tables.

The control below that used to select a file on the user's PC. Its appearance varies from browser to browser. The one on the right was produced by Firefox.



Upload a CSV File

Table: choose a table

File: Browse... No file selected.

Encoding: ● ANSI ○ UTF-8

☑ Convert case

☑ Trim spaces

☑ Delete existing rows

The file itself must obey a number of rules:

1. The name must have an extension of ".csv".
2. The decimal symbol used in floating point values must be a period or full stop, not a comma. This is because there is no reliable way to detect the locale of the user, so United States English is used throughout.
3. The field separator must be a comma, and not a semi-colon or tab.
4. The size must be less than 2MB. This is a PHP restriction (see below).

File encodings are discussed below. If the user has never heard of UTF-8, the chances are that his files are ANSI encoded.

The **Convert case** option converts the text on a field by field basis:

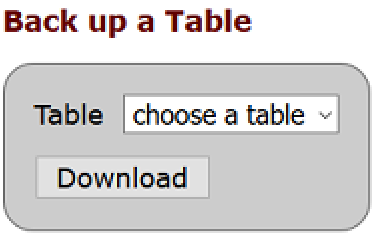
- Fields that are used as keys are converted to upper case. All the databases involved are case sensitive, and upper case references are recommended throughout. Postal codes/zip codes are also converted to upper case.
- Fields that are used as titles (names and addresses etc.) are converted to upper case words, e.g. Boston Crab Meat. This is sometimes called title case.
- Fields containing unstructured text, such as notes, are unchanged.

The **Trim spaces** option removes leading and trailing spaces from all fields. These are usually unnecessary, and can cause fields that contain key values to stop working properly. They can also play havoc with barcodes. Trailing spaces can sometimes arise as a result of the way that data is exported from other systems.

The **Delete existing rows** option clears the table before uploading the data in the CSV file. This is the default. Turning this option off allows data to be uploaded from more than one file, which may be necessary of the user hits the 2MB limit.

Back up a Table

This option is really only for long term backup purposes. The hosting service keeps daily backups for the last 14 days, so subscribers are usually well covered anyway.

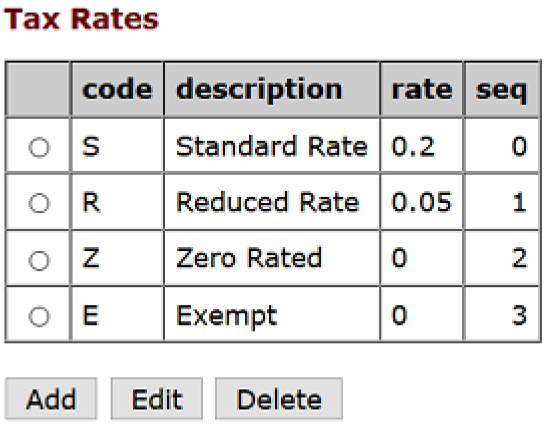


Back up a Table

Table: choose a table

Editing a Table

The web app provides basic services for viewing and editing tables. They are adequate, but limited when dealing with really large tables.



Tax Rates

	code	description	rate	seq
○	S	Standard Rate	0.2	0
○	R	Reduced Rate	0.05	1
○	Z	Zero Rated	0	2
○	E	Exempt	0	3

Downloading Orders

On the Orders menu, the **Download new orders** and **Clear change flags** options work in tandem. Every order record has a change flag, that determines whether it is a new record, or one that has already been processed. The change flag is set when an order is uploaded from a mobile device, and reset by the Clear option. The Download option downloads all new orders, i.e. ones that have the change flag set.

The usual sequence of operations is:

1. Download new orders (i.e. ones with the change flag set).
2. Process the orders in downloaded file.
3. Clear the dirty bits for the downloaded orders.

View an Order

This option allows the user to view a single order, with the header and details arranged nicely side by side. There is also a button that allows the user to download the order, in CSV format.

Duplicate Order Ids

This report shows all instances where the same order id has been used by more than one user. The database used by the web server allows this, but it is something that most companies would probably want to avoid.

Download Format

Orders are downloaded in denormalised form. The orders, orderDetails and products tables are joined together, and the result is downloaded. The records contain the following fields, in the order given:

Name of field	orders	orderDetails	products
employeeId	x	x	
orderId	x	x	
customerId	x		
orderDate	x		
requiredDate	x		
shipper	x		
freight	x		
shipName	x		
shipAddress	x		
shipCity	x		
shipRegion	x		
shipPostalCode	x		
shipCountry	x		
notes	x		
stamp	x		
custom	x		
type	x		
productId		x	x
name			x
unitPrice		x	
quantity		x	
discount		x	
taxCode		x	
taxRate		x	
custom		x	

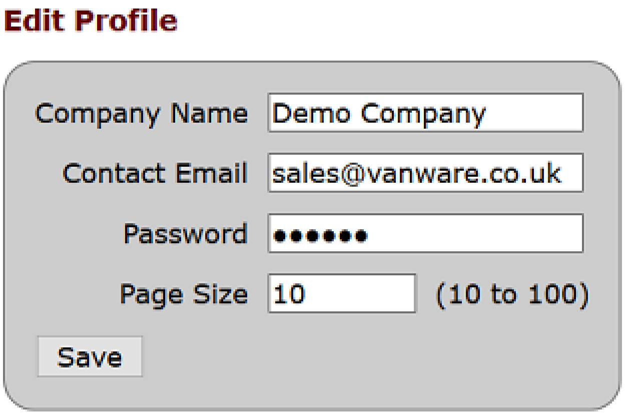
Customers

The Download and Clear options on the Customers menu work in the same way as the same options on the Orders menu, except that they work on the customers table rather than the orders table. The customers table also has a change flag. Users can upload customer records while in the field, either to create new customer records or to change the details of existing customers.

Edit Profile

The Edit Profile option allows the user to change some of the details stored on the company record.

The page size is the number of records that are displayed when a table is being edited.


Edit Profile

Company Name	Demo Company
Contact Email	sales@vanware.co.uk
Password	●●●●●●
Page Size	10 (10 to 100)

Sage™ Support

The hosting service for the Sales Manager allows subscriber to export orders from his company database into Sage Instant or Sage 50 Accounts, using the WebXel Order Importer for Sage. This is a third party product, which currently costs £100 per annum.

Once the Order Importer has been installed, it must be configured to use the hosting service as a data source. In particular, it requires two URLs:

1. The **import URL** is
   `http://www.vanware.co.uk/admin/sage1.php?companyId=`*nnn*`&password=`*xxxxxxxx*
2. The **notification URL** is
   `http://www.vanware.co.uk/admin/sage2.php?companyId=`*nnn*`&password=`*xxxxxxxx*

Here *nnn* is the company id, and *xxxxxxxx* is the password.

The first script returns an XML file, and the second one resets the corresponding change flags.

Self Hosting

Setup

Setting up a self-hosted server involves the following steps:

1. Download the scripts from the Vanguard Software website. Two sets of scripts are supplied: one for the PHP/MySQL platform and another for the ASP.Net/SQL Server platform.
2. Use the sql.txt file to create a database for the web server.
3. Populate the new database, including the employees table. For troubleshooting purposes, the sample data on the Vanguard Software website can be used for this purpose.
4. Modify the dbparams script to reflect this new database.
5. Create a folder for the scripts, somewhere in the HTML domain.
6. Copy the scripts into the new folder.

SSL

Sites with SSL certificates can use the

Data Transmission

Standard formats for transmitting data, such as XML and JSON, are quite verbose, and can typically increase the size of the data being transmitted by a factor of 4 or 5. This is not ideal when large volumes of data are being downloaded to a mobile device, especially when the user is paying for the bandwidth. Parsing such data may also be an issue, despite claims that are often made regarding the efficiency of this process.

Instead of using one of these standard formats, Vanguard Software chose to create their own format instead. The one that they chose is closely related to TSV (tab separated variables), except that x01 and x02 characters are used to separate fields and records, instead of TAB and LF (linefeed) characters.

Strictly speaking, the x01 and x02 characters are end of field and end of record markers, rather than separators. This means that an extra x01 character appears at the end of each record, and an extra x02 character appears at the end of the file.

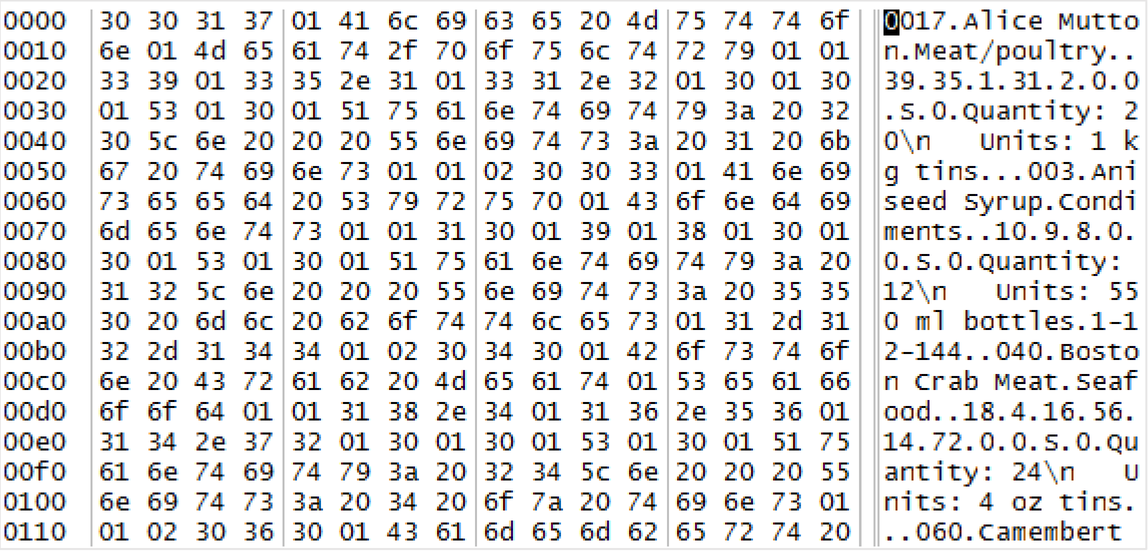
Troubleshooting

It is possible to download data from a web server using a standard web browser. For instance, to download the products table for the sample data, enter the following URL in the address bar (or just click on the link):

http:www.vanware.co.uk/server/products2.php?companyId=1&employeeId=1&password=password

A similar format can be used to download other data from other servers, providing the password is known. Most self hosted servers will probably not bother with the companyId parameter, and it is not used in any of the sample scripts.

In Firefox, the above downloads a file called products.php. This is confusing, because the file does not contain php code at all. It is actually the HTTP response to the query sent by the browser. An additional tool such is required in order to view the file properly.



```
0000  30 30 31 37 01 41 6c 69 63 65 20 4d 75 74 74 6f  0017.Alice Mutto
0010  6e 01 4d 65 61 74 2f 70 6f 75 6c 74 72 79 01 01  n.Meat/poultry..
0020  33 39 01 33 35 2e 31 01 33 31 2e 32 01 30 01 30  39.35.1.31.2.0.0
0030  01 53 01 30 01 51 75 61 6e 74 69 74 79 3a 20 32  .S.0.Quantity: 2
0040  30 5c 6e 20 20 20 55 6e 69 74 73 3a 20 31 20 6b  0\n   Units: 1 k
0050  67 20 74 69 6e 73 01 01 02 30 30 33 01 41 6e 69  g tins...003.Ani
0060  73 65 65 64 20 53 79 72 75 70 01 43 6f 6e 64 69  seed Syrup.Condi
0070  6d 65 6e 74 73 01 01 31 30 01 39 01 38 01 30 01  ments..10.9.8.0.
0080  30 01 53 01 30 01 51 75 61 6e 74 69 74 79 3a 20  0.S.0.Quantity: 
0090  31 32 5c 6e 20 20 20 55 6e 69 74 73 3a 20 35 35  12\n   Units: 55
00a0  30 20 6d 6c 20 62 6f 74 74 6c 65 73 01 31 2d 31  0 ml bottles.1-1
00b0  32 2d 31 34 34 01 02 30 34 30 01 42 6f 73 74 6f  2-144..040.Bosto
00c0  6e 20 43 72 61 62 20 4d 65 61 74 01 53 65 61 66  n Crab Meat.Seaf
00d0  6f 6f 64 01 01 31 38 2e 34 01 31 36 2e 35 36 01  ood..18.4.16.56.
00e0  31 34 2e 37 32 01 30 01 30 01 53 01 30 01 51 75  14.72.0.0.S.0.Qu
00f0  61 6e 74 69 74 79 3a 20 32 34 5c 6e 20 20 20 55  antity: 24\n   U
0100  6e 69 74 73 3a 20 34 20 6f 7a 20 74 69 6e 73 01  nits: 4 oz tins.
0110  01 02 30 36 30 01 43 61 6d 65 6d 62 65 72 74 20  ..060.Camembert 
```

Above is a dump of the first few bytes of data, produced using the hex editor Frhed.

The very first byte of the response is the status code, which in this case is 0 (good). The full list of status codes is as follows:

Value	Meaning
0	Good.
1	Unable to connect to database.
2	Not used.
3	Unable to execute query.
4	Invalid company id, employee id or password.

The first end of record marker (x02) is at offset x58. So the first record lies between offsets x01 and x57. The fields in the record can now be read off using the end of field markers (x01). This gives us productId=”017”, name=”Alice Mutton” etc. using the record layouts at the end of the manual.

The last field in the layout, called stock, is server side only, and so does not appear in the client side database. Being a multi-user system, stock levels are only monitored globally.

Upgrading

Whenever Vanguard Software changes a transaction, it creates a new version of it. This allows old versions of the app to continue using the old version of the transaction, while also allowing new versions to use the new one.

For instance, when a feature was added to allow companies to specify the colour of each pin on the customer map, a corresponding field was added to the customers table on the server. The transaction that was previously used to download the customers table was called customers2. This transaction was left unchanged, and a new transaction called customers3 was added.

The hosting service supports all versions of any given transaction, which allows it to support any version of the mobile app. However, the scripts available on the Vanguard Software website only support the transactions used by the mobile apps currently available on Google Play and the App Store.

Administrators of self-hosted servers have two possible upgrade strategies:

1. Lock down both the server, and the version of the mobile app. This only works if all the mobile devices run the Android OS. The APK file for the current version of mobile app can be downloaded from the Vanguard Software website. It should be kept carefully, and used to sideload new devices as they are brought into commission. Sideloading is a standard procedure for installing apps on Android devices, and the reader will find plenty of information about it on the internet.
2. Upgrade both the server and the mobile app, as new versions are released. This involves monitoring the Twitter feed for Vanguard Software for information about possible upgrades, and turning off automatic upgrades on the mobile devices. As new transactions are added to the scripts on the Vanguard Software web site, they can be patched in as necessary. Administrators of ASP.Net servers will have to be able to compile the project used by the aspx transactions.

In practice, most administrators prefer to lock the system down, rather than allow upgrades. Upgrades to the mobile app never destroy the existing data on a device, even when the database schema changes. The app keeps track of the database version and updates it automatically, as required.

IIS Setup

Introduction

These notes describe how to set up an ASP.Net/SQL Server web server in a Windows 10 environment. The procedures for earlier versions of Windows are similar.

Downloading Scripts

All the scripts required to set up a web server can be downloaded from the Vanguard Software website. This download has two folders:

1. php contains all the scripts for the PHP/MySQL environment.
2. aspx contains all the scripts for the ASP.Net/SQL Server environment.

The aspx folder contains the following:

1. bin is a folder containing the ASP.Net assembly for the web server. This folder should be copied to the IIS root directory.
2. Incs contains the source code for the VB.Net project that creates the ASP.Net assembly. Usually this can be ignored.
3. server is a folder containing the aspx scripts for the web transactions. These should be copied somewhere in the Documents folder.
4. sql.txt is a script that can be used to create the tables in the web server database,

The sample data should also be downloaded. This can be used later on to populate the web server, for test purposes.

SQL Server

The latest version of SQL Server Express (with tools) should be downloaded from the Microsoft website. The version with tools includes the management console, which is required. During installation, authentication should be set to Mixed Mode, as Windows Authentication may not always be possible. An SQL Server instance should also be created.

To set up a web server database:

1. Launch the SQL Server management console.
2. Create a database called Vanguard (easiest).
3. Run the SQL Server script called sql.txt, in the aspx folder. This will create all the tables used by the web server.
4. Run the Import and Export Data Wizard, to populate the tables with the sample data downloaded earlier.

In the aspx folder is a subfolder called server, which contains a number of ASP.Net scripts. Each script contains the string used to connect to the web server database. In order for the script to work correctly, this string should be changed to match the database that has just been created, e.g.:

```
cs = "Server=HP\SQLEXPRESS;UID=sa;PWD=master;"
```

Here the server name is HP\SQLEXPRESS, consisting of a computer name HP and an instance name SQLEXPRESS. The user id is sa, which belongs to the System Administrator. The password is master, which would have been supplied when instance was created.

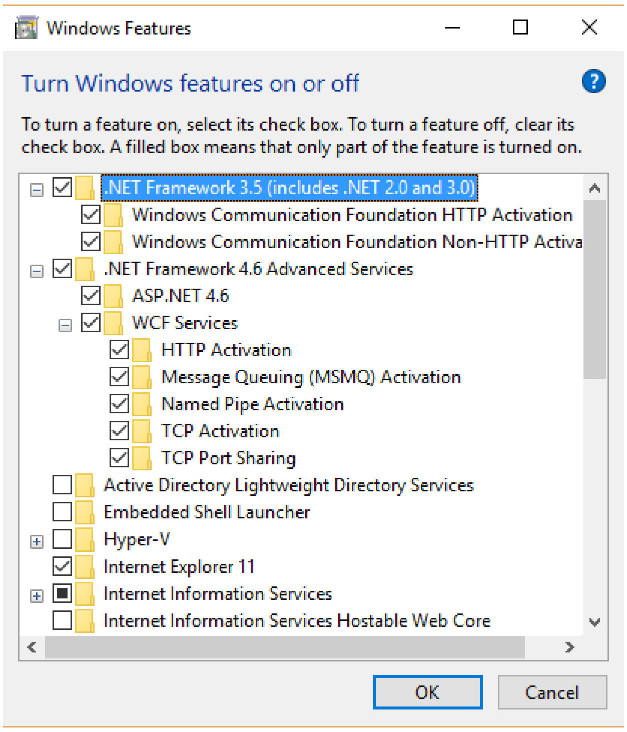
Installing IIS

On Windows 10, Internet Information Services (IIS) is not installed by downloading any sort of package from the Microsoft website.

Instead, the corresponding Windows feature is turned on. At the same time, The Windows Communication Foundation (WCF) options should be enable, for versions 3.5 and 4.6 of the .Net Framework.

To change the Windows features, click on Start > All apps > Windows System > Windows Console > Programs and Features > Turn Windows features on or off. The settings required to enable IIS are shown on the right.

Only the default features of IIS are required, together with all the WCF options, as shown on the right. Most web developers will also want Internet Explorer, which is on by default anyway. None of the other features are required.



Windows Features

Turn Windows features on or off

To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

⊟ ☑ .NET Framework 3.5 (includes .NET 2.0 and 3.0)
  ☑ Windows Communication Foundation HTTP Activation
  ☑ Windows Communication Foundation Non-HTTP Activa
⊟ ☑ .NET Framework 4.6 Advanced Services
  ☑ ASP.NET 4.6
  ⊟ ☑ WCF Services
    ☑ HTTP Activation
    ☑ Message Queuing (MSMQ) Activation
    ☑ Named Pipe Activation
    ☑ TCP Activation
    ☑ TCP Port Sharing
□ Active Directory Lightweight Directory Services
□ Embedded Shell Launcher
⊞ □ Hyper-V
☑ Internet Explorer 11
⊞ ■ Internet Information Services
□ Internet Information Services Hostable Web Core

Virtual Directory

The next step is to create a Virtual Directory. This is a regular directory on the hard drive, that maps into a directory in the HTTP space of the IIS server. To do this:

1. Launch the IIS Manager.
2. Right click on Default Web Site in the panel on the left.
3. Select Add Virtual Directory from the popup menu.
4. Enter an alias and a physical path, e.g. alias=server, physical path= C:\Users\Philip\Documents\My Web Sites\server.

The virtual directory test now appears in the first panel of the IIS Manager, and its contents are shown in the second.

Installing the Scripts

For every script in the virtual directory, read and execute permissions need to be granted to the Windows user groups Users and IIS_IUSRS. This is in addition to the change to the connection strings. The ASP.Net assembly IncsAssm.dll also needs to be copied to the bin folder in the default web site.

Testing

The web server can be tested using loopback, by entering a URL something like the following into the address bar of a browser:

```
http://127.0.0.1/server/products2.aspx?employeeId=1&password=password
```

Admin Database

Introduction

The Admin Database is a Microsoft Access application that can sync with a company database on the hosting service.

This is a great option if you are familiar with Microsoft Access, and you want to use it to build your own back end services.

This chapter assumes that you are familiar with Microsoft Access, and uses some of its key concepts without explanation.

Installation

To install the Admin database, just download the admin.zip file from our web site, and unzip it into your documents folder.

The admin folder contains:

1. **Admin.mdb** is the application. It contains all the Microsoft Access objects, but no data.
2. **Blank.mdb** and **Sample.mdb** are databases that Admin.mdb can link to. They contain data, but no Microsoft Access objects.
3. **logo.ico** is just the icon file for Admin.mdb.

All the files have to remain in the same folder. There is no formal installation procedure, but it is easy enough to add a link to the admin.mdb file to your Start menu.

VBA module

The VBA code for the admin database was specifically designed with portability in mind, and it should run on any version of Windows and Microsoft Access.

There are no library references, and all objects are weakly typed. You may want to change this if you are writing your own VBA procedures.

The code uses DAO whenever it can. However it uses ADO to update the linked tables, as this is the only way to get an updatable recordset.

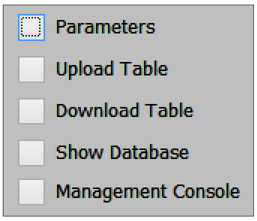
Every procedure begins with the lines

```
Option Compare Database
Option Explicit
```

This ensures that all variables are declared, and makes string comparisons case insensitive. These are fairly standard VBA conventions.

Switchboard

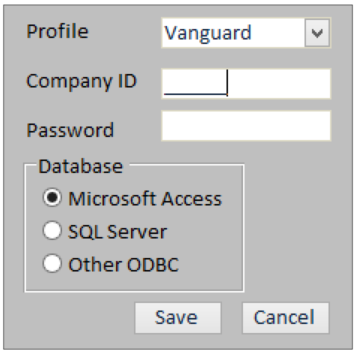
The admin database uses a minimal switchboard, with just five options. The last two are convenience functions to open the navigation pane and the login screen for the web app, respectively.



The switchboard uses a custom form, and only has one page. To edit the switchboard, you will have to install the switchboard manager, which is not installed by default.

Linked Database

The admin database follows the standard design practice of storing the Microsoft Access objects and the tables in separate databases. This makes it easier to manage the data, and to upgrade the application.



Profile: Vanguard

Company ID:

Password:

Database

● Microsoft Access

○ SQL Server

○ Other ODBC

The Parameters form specifies where the app should look for its linked tables.

The profiles are contained in the Profiles table. To add or remove profiles, edit the table directly.

The profile points to the database that contains the tables that the application links to, and this depends on the type of database being used.

For Microsoft Access databases, the profile is the name of the mdb file, which must be contained in the same folder as admin.mdb. If it does not exist, it will be created.

For other types of database, the profile is the name of a user DSN. DSNs (Data Source Names) are pointers to ODBC databases, and are created by the ODBC Data Source Administrator. This is an applet in the Administration Tools section of the Control Panel.

For ODBC databases, the database must be created manually. For SQL Server databases, the easiest way to do this is to open the blank.mdb database, and use the SQL Server migration tool in the toolbar.

The main advantage of using an SQL Server database instead of a Microsoft Access one is that it works better in a multi-user environment. You can pit the database on a server and have more than one user work with it at the same time.

Once you have set up your database, you will be able to use the forms provided to upload and download data, in the same way as you can with the web app for the hosting service. But you will also be able to add additional features to suit your particular business needs.

Macro Security

No discussion of Microsoft Access would be complete without mentioning its nemesis, which was macro security.

When the first malware arrived in the year 2000, Microsoft was caught out by the sudden need to tighten up the security of its Office suite of applications.

Its reaction was to effectively ban Office documents that contained VBA code. For Word documents and Excel spreadsheets that was a reasonable response, but for Access databases it was catastrophic.

It means that when you open the admin database for the first time, you will probably encounter an error message, saying that there has been a macro security violation.

This has nothing to do with Access macros. It is actually referring to the VBA project. For other Office documents, this typically contains macros.

In the long run, the only way to get any work done is to turn off Macro Security for Microsoft Access databases.

To do this:

1. Open any Microsoft Access database.
2. Click on **File > Options > Trust Center > Trust Center Settings**.
3. On the Trust Center form, click on **Macro Settings >Enable all macros**.
4. Close the database.

Database

Naming Conventions

The Sales Manager database conforms to the standard for Android SQLite databases. This stipulates that the names of all tables and fields should be in camel case, e.g. taxCode instead of tax_code or whatever.

Referential Integrity

None of the relationships shown in the above diagram is enforced in any way. The mobile app is designed in such a way that this does not really matter. It is up to the user to ensure that there is no missing data.

Standards

The Sales Manager database does not conform to the normal standards for relational databases. In particular, the price fields are all rolled up into the products table, rather than having separate tables for price lists and for prices. This was done in order to make the database easier for the user to manage.

Duplicate Order Ids

Note that the employeeId field is part of the primary key for both the orders and orderDetails tables. This prevents users from overwriting each other’s orders accidentally.

Orders

Note that orders are stored in two tables: one for the order headers and another for the detail lines. This is standard design practice.

Business Considerations

Restricting Users

On the Android platform, it is possible to restrict the actions that certain users can perform. Restricted users can download data from the web server, take orders and upload them back to the web server. But they cannot change any of the company data stored on their mobile devices, including their own user settings.

A user is restricted by setting the restricted flag in his employees record, on the web server. Then the Check option must be selected on the Downloads page of his mobile device. This option reads the record and restricts the device. The device will then remain restricted until the flag is reset on the server.

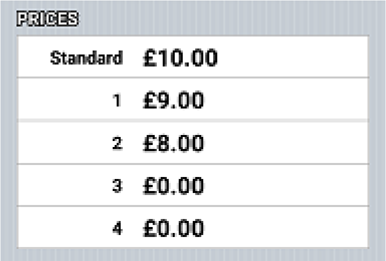
So to restrict a mobile device for a given user:

1. Set the flag on the web server.
2. Take his device.
3. Initialise his user settings.
4. Go to the Downloads page.
5. Select the Check option.
6. Hand his device back to him.

This is a cumbersome process, and it cannot be administered directly, but it does work.

Prices

The Sales Manager supports up to five price lists. These are called Standard, 1, 2, 3 and 4. Each product is assigned a price in each of the price lists, as shown on the right. The sample data only uses three price lists, and the prices in the last two price lists are set to zero.



PRICES

Standard	£10.00
1	£9.00
2	£8.00
3	£0.00
4	£0.00

Each customer and order is assigned a price list. When a new order is create for a given customer, the price list for that order is set to the price list for the customer. When a new detail line is added to an order, the price list for the order is used to select which price to use.

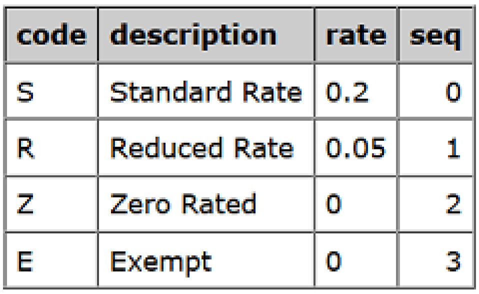
The prices shown above are for Aniseed Syrup. Suppose that a given customer orders from price list 1, and that a new order is taken for that customer. If the customer orders Aniseed Syrup, he will be charged £9.00 per unit, because that is the price in list 1.

Tax Rates

The Sales Manager has a very flexible approach to tax rates, designed to accommodate as many tax regimes as possible:

1. Tax rates can be applied to both customers and products.
2. Individual customers and products may have no tax rate assigned to them.
3. When they are created, orders are assigned the same tax rate as their customers.
4. A tax rate for an order overrides the tax code for a product.

If a line is added to an order that already has a tax rate, the tax rate for the product is ignored. This allows the system to support customers who are exempt from paying tax on any product.



code	description	rate	seq
S	Standard Rate	0.2	0
R	Reduced Rate	0.05	1
Z	Zero Rated	0	2
E	Exempt	0	3

Each tax rate consists of a code, a description, a rate and a sequence number. The tax rates for the sample data are shown on the right. The unique key is actually the code, but it is more convenient to think of them as rates.

The sequence number determines the order in which the tax rates are displayed and printed. The first rate in this sequence is the standard rate. This is the rate that is applied if no tax rate is applied to either the order or to the product.

When a line is added to a product, both the code and the rate are copied to the detail record. This allows tax rates to changed, without affecting existing orders.

Usually a company will assign tax rates to either customers or to products, but not both. Assigning tax rates to customers suits the US sales tax model, while assigning tax rates to products is more suited to the EC system of Value Added Tax (VAT).

Business model

The Sales Manager has a flexible business model that should suit most users in the US and the EU, and will work for many users in other countries as well.

1. All sales are in a single currency, which is the one for the current locale.
2. There are multiple rates of sales tax, which can be specified per customer or per product.
3. Each customer is charged according to one of five price lists.
4. A separate discount can be applied to each order line.
5. Some customers can place orders with any user. Others can only order from a given user.
6. Some customers are charged special prices for certain products.
7. Prices and tax rates can vary over time.
8. Products can become obsolete.
9. Once an order has been placed, it cannot be changed.

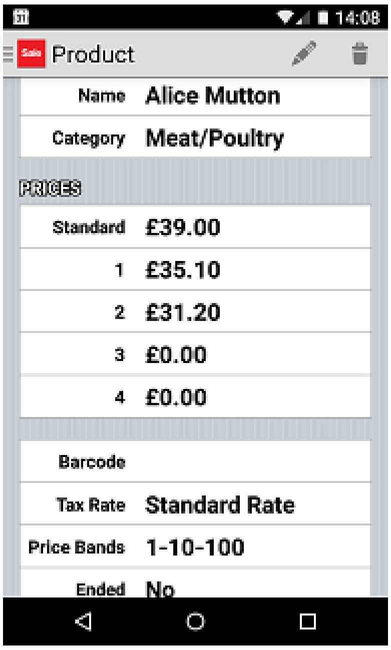
Banding

Banding is a simple mechanism for pricing a product based on the quantity ordered. In the first screenshot on the right, banding has been defined for the product Alice Mutton (product id 017).

The string “1-10-100” in the price bands field means that the user can order 1 unit at the standard price, 10 units at the price in price list 1, or 100 units at the price in price list 2. These quantities are obtained by splitting the price bands string, using the character “-” (minus sign) as separator.

These quantities and prices summarised in the table below:

Quantity	Price
1	£39.00
10	£35.10
100	£31.20

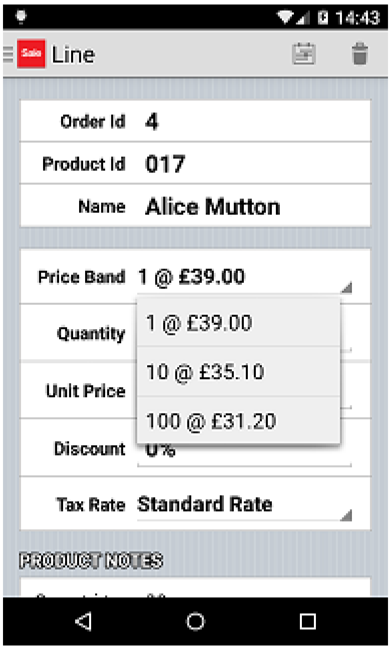


Name	Alice Mutton
Category	Meat/Poultry

PRICES

Standard	£39.00
1	£35.10
2	£31.20
3	£0.00
4	£0.00

Barcode	
Tax Rate	Standard Rate
Price Bands	1-10-100
Ended	No



Order Id	4
Product Id	017
Name	Alice Mutton

Price Band	1 @ £39.00
Quantity	
Unit Price	
Discount	0%
Tax Rate	Standard Rate

1 @ £39.00
10 @ £35.10
100 @ £31.20

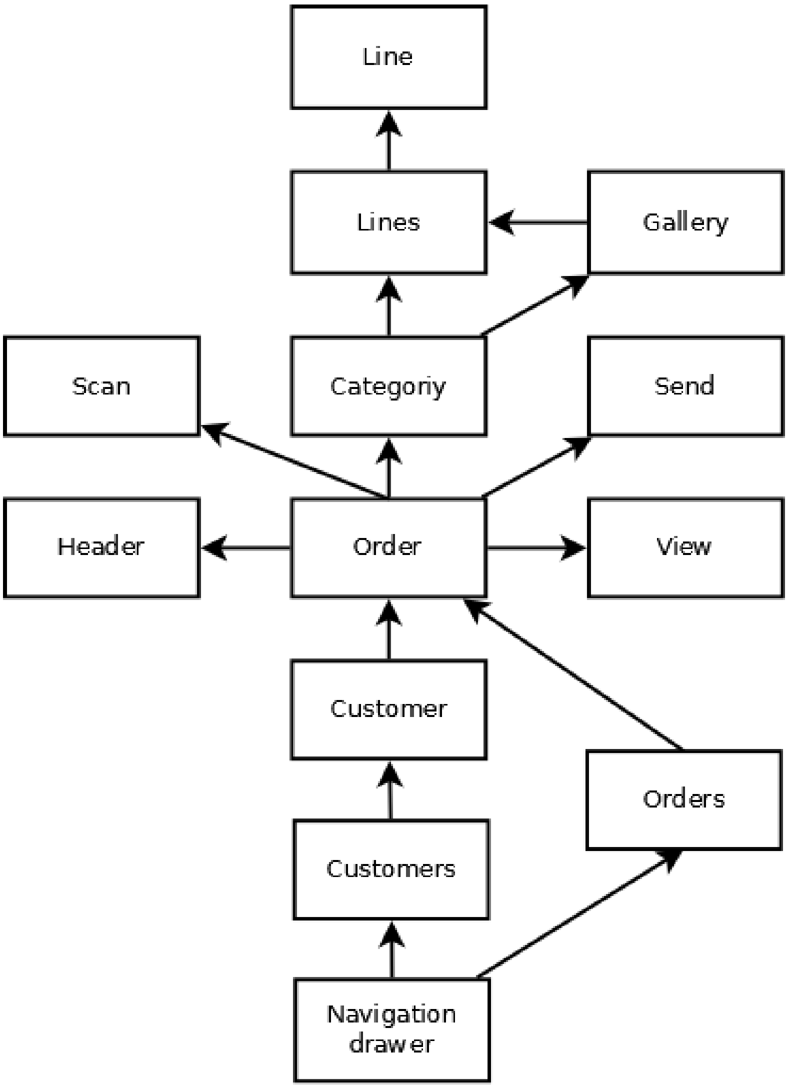
PRODUCT NOTES

When the product is added to an order, the Line screen contains an extra field, a spinner (drop-down list) called Price Band. Tapping this displays the three price bands as options, as shown in the second screenshot.

When the user selects one of the options in the list, both the quantity and unit price fields are filled in. For instance, selecting the second option would set the quantity to 10, and the unit price to £35.10. Banding is not available for restricted users.

Screens

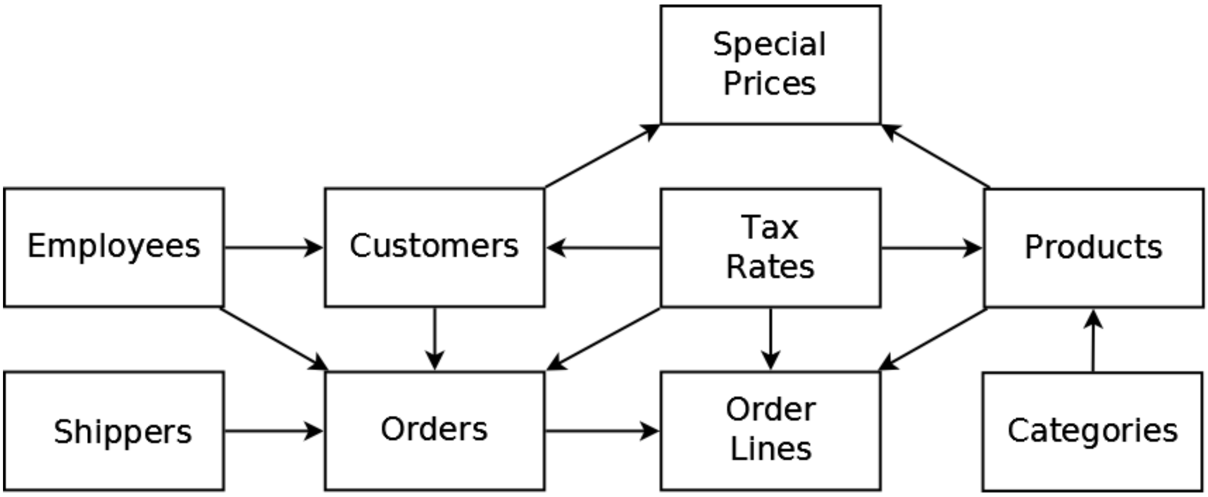
Taking an Order



The diagram above shows the main screens used to take an order, arranged hierarchically.

Schema

Diagram



The above diagram shows the relationships between the tables in the database used by the web server. The one used by the mobile app is almost identical, except that there is no employees table, and no employeeId fields in the other tables.

Record Layouts

Introduction

The record layouts below are for the tables in the Sales Manager database. They were produced by a VBA module, using remarks stored in a Microsoft Access version of the database.

categories

Name	Type	Description
category	Text	Name of category. Uses a breadcrumb format.

custom

Name	Type	Description
section	Integer	Section number: 0=order header, 1=detail line.
seq	Integer	Order in which control appears within section.
type	Integer	Control type: 0=header, 1=spinner, 2=checkbox, 3=text.
name	Text	The name of the control, as shown to the user.
vals	Text	Any additional values. For spinners, this is a list of options, separated by commas.

customers

Name	Type	Description
customerId	Text	Unique id of customer.
employeeId	Long	Employee who sells to customer. If zero, any employee can sell to customer (server side only).
priceList	Byte	Price list for orders. Integer between 0 and 4.
taxCode	Text	Tax code for orders. If blank, customer uses tax codes on product records.
name	Text	Name of customer.
address	Text	Street address.
city	Text	
region	Text	
postalCode	Text	post code or zip code.
country	Text	
latitude	Long	In degrees multiplied by 10^6. Zero if location is not known.
longitude	Long	In degrees multiplied by 10^6. Zero if location is not known.
contact	Text	Name of contact.
contactTitle	Text	Job title of contact.
phone	Text	Phone number.
email	Text	Email address.
notes	Memo	Notes about the order.
discount	Double	default value for new orders.*new
colour	Integer	0=red, 1=yellow, 2=orange, 3=green.*new
dirty	Boolean	Set when record is uploaded by a mobile device (server side only).

Employees

Server side only.

Name	Type	Description
employeeId	Long	Unique id of employee.
name	Text	Name of employee, for use by system administrator.
password	Text	Password for web server.
restricted	Boolean	If set, user cannot change any company data.

orderDetails

Name	Type	Description
employeeId	Long	Unique id employee who created order (part of primary key, server side only).
orderId	Long	Unique id of order (part of primary key).
productId	Text	Unique id of product (part of primary key).
unitPrice	Currency	Unit price. Copied in case price changes.
quantity	Double	
discount	Double	
taxCode	Text	tax code.
taxRate	Double	tax rate. copied in case rate changes.
custom	Text	data for custom fields on Edit Line screen.

orders

Name	Type	Description
employeeId	Long	Unique id employee who created order (part of primary key, server side only).
orderId	Long	Unique id of order (part of primary key).
customerId	Text	Unique id of customer.
orderDate	Text	Date of order.
requiredDate	Text	Date when order is required.
shipper	Text	Delivery method.
freight	Currency	Delivery charge.
shipName	Text	Name of recipient.
shipAddress	Text	Delivery address.
shipCity	Text	
shipRegion	Text	
shipPostalCode	Text	
shipCountry	Text	
notes	Memo	
stamp	Text	Timestamp set by server when order was uploaded.
custom	Text	Data for custom fields on Edit Header screen.
type	Integer	0=invoice, 1=credit note, 2=quote.
dirty	Boolean	Set when record is uploaded by a mobile device (server side only).
latitude	Long	In degrees multiplied by 10^6. Zero if location is not known.
longitude	Long	In degrees multiplied by 10^6. Zero if location is not known.

products

Name	Type	Description
productId	Text	Unique id of product.
name	Text	Name of product.
category	Text	Product category.
barcode	Text	
price	Currency	Price shown on Products screen.
price1	Currency	
price2	Currency	
price3	Currency	
price4	Currency	
taxCode	Text	Tax code. Can be overridden by tax code on order.
discontinued	Byte	If true, product is not shown on Pick Lines screen.
notes	Memo	
banding	Text	Price banding, e.g. 1-10-100.
stock	Double	Current stock level (server side only).

shippers

Name	Type	Description
shipper	Text	Name of shipping method.

specialPrices

Name	Type	Description
customerId	Text	Unique id of customer.
productId	Text	Unique id of product.
price	Double	Special price.
description	Text	Reason for awarding special price.

taxRates

Name	Type	Description
code	Text	Code printed on invoices.
description	Text	Description of tax rate.
rate	Double	Rate as percentage.
seq	Long	Sequence in which rate appears.