

User Interface Style Guide

For J2ME™ Applications Running on Motorola iDEN™ Handsets

***i*DEN**



Version R1.0

1 Table of Contents

1	Table of Contents	2
2	Table of Figures	4
3	Introduction.....	5
3.1	Audience	5
3.2	Reference Documents	5
3.3	Acronyms & Definitions.....	6
3.4	Conventions used in this Document	7
4	General Design Considerations.....	8
4.1	Good User Experience	8
4.2	User Control	8
4.3	Responsiveness.....	8
4.4	Customization.....	8
4.5	Ease of Use.....	8
4.6	Intuitive	9
4.7	Consistency	9
4.8	Clarity	9
4.9	Aesthetics	9
4.10	Forgiveness	9
4.11	Resources	9
4.12	Documentation	10
4.13	Networking Applications	10
5	iDEN J2ME Capable Handset Controls	11
5.1	Key guide	11
5.2	General.....	11
5.3	Keypad	12
5.4	Soft Buttons.....	12
5.5	Radio Buttons	13
5.6	Check Boxes.....	13
5.7	Gauge	13
5.8	Scrolling.....	14
5.9	Text Fields.....	14
5.10	Lists.....	14
5.11	Tables.....	15
5.12	Function Menu	15
5.13	Games	15
6	iDEN J2ME Capable Handset User Interface.....	16
6.1	High Level API & Low Level API.....	16
6.2	General UI considerations.....	16
6.3	Screen Refresh.....	17
6.4	Dialogs	18
6.5	Application Launch	18
6.6	Title Bar	18
6.7	Tickers.....	19
6.8	Main Menu	19

6.9	Soft Button indicators	20
6.10	Data entry fields	21
6.11	Scroll Bars.....	22
6.12	Data Listing/Viewing.....	22
6.13	Notifications	23
6.14	Error Messages	23
7	Contact Information	24

2 Table of Figures

Figure 1 - iDEN Handset Keys and Soft Buttons	6
Figure 2 - iDEN Handset Key Guide	11
Figure 3 - Soft Buttons	12
Figure 4 - Radio Buttons	13
Figure 5 - Check Boxes	13
Figure 6 - Gauge	13
Figure 7 - Text Fields	14
Figure 8 - Lists	14
Figure 9 - Title Bar.....	18
Figure 10 - Tickers.....	19
Figure 11 - Main Menu	19
Figure 12 - Soft Button Indicators	20
Figure 13 - Data entry fields.....	21

3 Introduction

This document has been developed in partnership between Motorola and Quality Partners. This document is intended as a guide for designing a consistent, user-friendly interface for J2ME™ applications running on the Motorola iDEN™ Handsets. It provides guidelines and principles governing good user interface design as well as specific standards for the iDEN J2ME Capable Handsets. This guide discusses in detail the various screen elements and controls of the application. The intent of this document is not to outline all the requirements for compliance with the Motorola Application Certification Program. These requirements can be found in the Motorola™ Application Certification Program Developer Handbook for J2ME™ applications running on the Motorola iDEN™ Handsets.

3.1 Audience

This document is intended for developers interested in creating applications for the Motorola iDEN J2ME Capable Handsets.

3.2 Reference Documents

Introduction to Wireless Programming with the MID Profile, Sun Microsystems, WWW
<http://java.sun.com/aboutJava/communityprocess/final/jsr037/index.html>

Java™ 2 Platform Micro Edition (J2ME™) Technology for Creating Mobile Devices, White Paper, Sun Microsystems, May 19, 2000.
<http://developer.java.sun.com/developer/technicalArticles/wireless/midp/index.html>

Motorola Application Certification Program Developer Handbook for iDEN J2ME Capable Handsets, Product Quality Partners, Inc, August 23, 2001.
<http://www.qpqa.com/motorola/iden/index.html>

Mobile Information Device Profile (JSR-37) JCP Specification 1.0, Sun Microsystems, September 1, 2000.
<http://java.sun.com/aboutJava/communityprocess/final/jsr037/index.html>

Motorola J2ME SDK Components for the J2ME Platform, Motorola
<https://commerce.motorola.com/idenonline/ideveloper/index.cfm>

*(C) Copyright 2001 Motorola, Inc. All rights reserved.
Motorola and iDEN are trademarks of Motorola, Inc.
J2ME is a trademark of Sun Microsystems, Inc.*



Figure 1 - iDEN Handset Keys and Soft Buttons

3.3 Acronyms & Definitions

3.3.1 J2ME

Java 2 Micro Edition™ (J2ME™) is an edition of Sun Microsystems, Inc.'s Java technologies for small, resource-constrained devices.

3.3.2 iDEN

integrated Digital Enhanced Network™ (iDEN™) is a digital technology for wireless communication developed by Motorola that divides a channel into multiple “slots”. Each slot can carry one voice or data transmission. Thus, multiple services can be delivered from a single, integrated wireless communications system. With iDEN technology, Motorola brings together dispatch radio, full-duplex telephone interconnect, short message service and data transmission into one network.

3.3.3 CLDC

Connected, Limited Device Configuration™ (CLDC™) is one of the two configurations within the J2ME architecture. It is intended for devices that have very simple user

interfaces (compared to desktop computer systems), minimum memory budgets starting at about 128 kilobytes, and low bandwidth, intermittent network connections. CDLC defines the minimum set of Java virtual machine features and Java class libraries that device manufacturers and developers can assume are available on a particular “category” of devices.

3.3.4 *MIDP*

Mobile Information Device Profile™ (MIDP™) defines the minimum set of Java Application Programming Interfaces (APIs) available on a particular “family” of devices, such as mobile handsets and two-way pagers. Profiles are implemented upon a particular configuration and applications are written for a particular profile.

3.3.5 *Soft Button indicators*

The soft button indicators consist of 3 labels that appear on the bottom row of the LCD display. These labels indicate which button to press to activate certain functionality of the application.

3.3.6 *Keys*

The keys are all of the buttons that appear on the front face of the handset.

3.4 **Conventions used in this Document**

3.4.1 *Should*

This means that there may exist valid reasons in particular circumstances to ignore a particular requirement in this document, but the full implications of ignoring this requirement **MUST** be understood and carefully weighed before choosing a different course.

3.4.2 *Must*

Where the document says **Must**, this indicates that the information is a requirement for compliance with Motorola Standards as part of the Motorola Application Certification Program for the iDEN J2ME Capable Handsets. For more information about the Certification Program, please refer to the Developer Handbook available from the Quality Partners web site (www.qpqa.com/motorola/iDEN)

4 General Design Considerations

The iDEN J2ME Capable handset supports the J2ME (Java 2 Micro Edition) Mobile Information Device Profile, or MID Profile. This profile is designed to operate on top of the J2ME Connected Limited Device Configuration (CLDC). The following are some basic design considerations that should be followed by all iDEN J2ME Capable handset applications.

4.1 Good User Experience

The aim of good UI design is to give the user a good experience. This is similar to the way most Windows and Apple applications behave, in that there is a consistent UI across applications. The application should allow the user to perform the necessary tasks with a minimum of effort. The user should be made to feel good about using the application. Care should be given to avoid leaving the user feeling frustrated. The goal is to have a consistent user experience across all applications. The same keys should deliver the same results, e.g. all applications should allocate identical functionality to the same soft buttons.

4.2 User Control

The user should always be in control of the application.

4.3 Responsiveness

The application should allow the user to quickly move between different areas of the application. Too long a delay between the user performing an action and the application displaying the result does not confer a good user experience. Feedback is important to the user. When a given action is performed, presenting feedback to inform the user that the action was carried out is important. The response time of this feedback is also important. If the task being performed takes a long time, a “Please Wait” or similar dialog should be displayed.

4.4 Customization

The user should be able to customize the UI to their needs, where the application design and size or resource limitations permit. The application should provide default controls that are common to most users.

4.5 Ease of Use

The application should make the users task easier. The user should not need to spend too much time figuring out how to navigate the user interface, and be able to perform the task at hand with a minimum of confusion.

4.6 Intuitive

The interface should be intuitive, and straightforward. A good way to achieve this is using the object-action method. Allow the user to select an object and perform an action on the given object (such as selecting a memo, and editing the memo).

4.7 Consistency

The application should be consistent in terms of the user interface throughout the application. Where controls are used for specific functionality in one area of the application, these controls should be used for the same functionality in all areas of the application. The application should also be consistent with other applications running on the same platform, to give the user a degree of consistency across the platform as a whole.

4.8 Clarity

The application should aim for visual and conceptual clarity. The user interface should be clear to the user, and not be visually confusing. This is especially important, as the screen size is relatively small. The application should not clutter the screen with controls, unnecessarily. Dialogs and messages should be clear and to the point conveying the appropriate information in as simple a method as possible. Messages should not just indicate what the user has done wrong, but also what the user needs to do to correct the problem, if the screen real estate permits.

4.9 Aesthetics

While the application needs to be visually pleasing to the user, a simple design is also necessary. As the screen size and system resources are limited, overuse of flashy design components, such as flashing cursors, or three-dimensional graphics is unnecessary. A simple design is sufficient to provide the user with an aesthetically pleasing user interface.

It is important to reduce the amount of changing information. Keep the amount of changing information to one item where possible, and keep the users focus limited to one area of the screen at a time. When presenting a lot of information to the user, ensure the most important information is displayed first.

4.10 Forgiveness

It is important to give the user an option to undo actions, especially if that action can result in deleting data. For example, deleting a memo item should display a message warning the user that the memo will be deleted.

4.11 Resources

For network-enabled applications, when data transmission has completed, the application should release the port and allow basic phone services to resume.

The application .jar file should be no larger than 50kb (compressed size).

Note: For additional information regarding functional requirements, please see the Motorola Application Certification Program Developer Handbook for the iDEN J2ME capable Handsets.

4.12 Documentation

Due to the limited available space for running applications, it is recommended that the help within the application be minimized. This information can be included in electronic form accompanying the application. This documentation must be in .pdf or .html format.

The accompanying documentation or users guide should contain sufficient information to inform the user of how to operate the application. The user guide should not assume the user has any prior knowledge of the use of similar applications.

It is recommended that the user guide provide detailed instructions on the use of the applications functionality. Providing screen-shots can clearly indicate to the user the functionality of certain aspects of the software.

When referring to the navigation key, the recommended terms are “Navigation Key” or “Rocker Key”.

4.13 Networking Applications

The iDEN handset cannot transfer phone data and packet data at the same time. When a network enabled application is transmitting packet data, the phone services will be blocked. The application should warn the user that phone services will be blocked when network data is being transmitted. This could be in the form of a dialog or message that is displayed upon launch of the application.

The user documentation must explain that phone services will be blocked when packet data is being transmitted and that packet data service will incur a cost to the user.

5 iDEN J2ME Capable Handset Controls

5.1 Key guide

The following figure shows the keys on the handset, and gives a guide to the keys as referenced in this guide:



Figure 2 - iDEN Handset Key Guide

5.2 General

The application should give clear instructions to the user as to the controls that can be used. This may be through user documentation, or using online help screens.

5.3 Keypad

5.3.1

It should be clear from the application screen, of the function of all the various keys.

5.3.2

The use of the keypad and other keys should be consistent throughout the application.

5.4 Soft Buttons



The soft buttons are used to activate certain functionality within the application. The functionality activated by the soft buttons is indicated by the soft button indicators at the bottom of the screen.

Figure 3 - Soft Buttons

5.4.1

The functionality of the soft buttons must be consistent throughout the application. For example, if the left soft button is used to cancel an operation in a particular dialog, this should be used for the same function throughout the application.

5.4.2

Where an action performed will result in deletion or destruction of data, the left soft button must be used to indicate negative confirmation, e.g. No, Back or Cancel. The right button should be used to indicate positive confirmation, e.g. Yes, Go, Select.

5.4.3

The following button combinations should be used to give the user consistency across different applications:

Left Soft Button	Middle (Menu) Soft Button	Right Soft Button
Cancel	Menu	OK/Done
Back		Next
Exit		Select/Deselect
		Start
No		Yes
		View

Note that for critical confirmation or double confirmation dialogs, the behavior is different. The first confirmation is always reversed and the second confirmation is

reversed again. This is to prevent the user from executing a potentially damaging function by mistake. e.g. For the Master Reset function, the first response requires the left soft key to be pressed to answer yes to the Master Reset. The subsequent confirmation requires the user to press the right soft key to accept (or OK) the function.

5.5 Radio Buttons



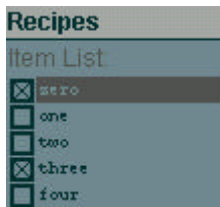
A radio button field allows the user to make a single choice from a list of items.

Figure 4 - Radio Buttons

5.5.1

Where multiple fields are arranged in a vertical order, selection of radio buttons should be accomplished by the up or down navigation keys, and pressing the SEND button to check/uncheck the box.

5.6 Check Boxes



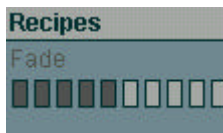
Check boxes allow a user to select one or more items from a list.

Figure 5 - Check Boxes

5.6.1

Selection of check boxes may be accomplished by pressing the SEND key. This needs to be clearly indicated to the user.

5.7 Gauge



A Gauge allows the user to visually select a value from a range of values.

Figure 6 - Gauge

5.7.1

Gauges may be modified by using the left/right navigation key.

5.8 Scrolling

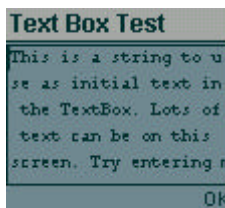
5.8.1

When scrolling through list items, the items should rotate in a cyclical manner. (i.e. moving down from the last menu item should return the highlight to the first item in the list.) The scrolling must use the up/down navigation “rocker” keys.

5.8.2

Where horizontal movement is possible, the left/right navigation key should be used.

5.9 Text Fields



Text fields can be used to allow the user to enter text.

Figure 7 - Text Fields

5.9.1

Data entry should use the left/right navigation key to move the cursor through the data entry field.

5.9.2

Data entry should use the numeric keypad as indicated on the keys and the user manual. The sequence of cycling through the number and letters for each key should be consistent with the characters printed on the appropriate key.

5.10 Lists



A list presents a number of items to the user. These items could be records in the application, or functions that can be selected.

Figure 8 – Lists

5.10.1

Lists should be navigated by the up/down navigation key, where a list allows for horizontal navigation, the left/right navigation key should be used.

5.11 Tables

5.11.1

Navigating in a table should be performed using the navigation key.

5.12 Function Menu

5.12.1

The function menu should be accessible using the center menu button.

5.13 Games

5.13.1

The controls used for games, should be easily accessible, such that the user can access all the required controls while maintaining a comfortable grip on the handset

5.13.2

Where vertical and horizontal movement is required, the navigation key should be used.

6 iDEN J2ME Capable Handset User Interface

6.1 High Level API & Low Level API

The high level API includes classes that can be called to display User Interface controls such as text boxes and radio buttons. The behavior of these controls is determined by the classes themselves and the developer has little control over the look and feel of the controls. When using the high level API's, some of the recommendations given in this guide may not be possible. Some of the recommendations may already be implemented in the high level API's; the recommendations are repeated here, as a reference for developers using the low level API's.

When using the low level API, the developer has more control over the screen, and input mechanisms. Some applications may incorporate controls using low level API classes that resemble the controls available with the high level API's. In these cases it is recommended that these controls behave consistently with the high level API controls.

6.2 General UI considerations

6.2.1

Abbreviations should be sensible and unique. Try to avoid long abbreviations that are not commonly understood. If it is necessary to use a significant number of abbreviations, a glossary may be included in the user guide.

6.2.2

Where an action to be performed may result in data-loss, a dialog must be displayed indicating that data-loss will occur, and the user should be given an option to cancel the current operation.

6.2.3

Where a license expiration or registration screen is automatically displayed, this must appear only once upon expiration of the license. The license expiration message must not display more than once throughout the life of the application. If the application disables functionality when the license expires, the application may display a message, when disabled functions are attempted, informing the user that the function is not currently available. This message may give the user an option to enter an activation code at this point. If the developer wishes to track the number of remaining or used sessions or days and provide this information for the user, the application may display this information in an "About" dialog, or in a specific user selected menu option.

6.2.4

The screen content must be readable to the naked eye (i.e. 20x20 vision). Care must be taken to avoid visually confusing screen content, such as selection, scrolling and text highlighting. All elements must provide appropriate contrast for readability.

6.2.5

If the application displays an icon, the icon should be unique to the application. An application can reuse standard Motorola icons only where they represent the same

functionality as the corresponding handset icon. For example, the “Menu” icon can be used in an application to indicate that pressing the “Menu” key will pop up an application menu. To be consistent with standard use, when the Menu key is pressed, the Icon should be “highlighted.” The icon in the entry screen should fit in an 11 pixel square area. The icon should be in the upper left corner of the screen. The icon must be clean, visible and appropriate to the application. Care should be taken to avoid icons that do not clearly represent the application, or icons that are not easily distinguished.

6.2.6

The application must not contain inappropriate or offensive text. The application must not include any material or information that is fraudulent, misleading, obscene, pornographic, defamatory, trade libelous, libelous, slanderous, unlawfully harassing or injurious, excessively violent, in violation of personal or property rights, regulation or law, or otherwise unfit for publication.

6.2.7

The data space and program space requirements must be present in the .jad file. These values must match the values present in the documentation and the about screen from the Java menu on the handset.

The data and program space values must use the following syntax (n is the value in Kbytes):

“iDEN-Program-Space-Requirement:n”

“iDEN-Data-Space-Requirement:n”

6.3 Screen Refresh

6.3.1

When moving from one screen to another, or when displaying or dismissing dialog boxes, the screen must refresh cleanly, without leaving residual graphics from the previous screen or dialog.

6.3.2

When the screen content is changed, this change must be smooth, and must avoid flickering, or slow screen changes.

6.3.3

The screen must not be refreshed unnecessarily. Flickering, corrupted screen or bleed through must be avoided.

6.3.4

Animations within the application should be smooth, and clearly visible. Where these animations are as a result of user input, the control to movement should be smooth.

6.4 Dialogs

A dialog box is a window that displays information, or a prompt for user input. A dialog box may be full screen or a window within the screen.

6.4.1

Where text appears in dialog boxes, this text must be clear and not truncated.

6.5 Application Launch

6.5.1

If the application takes longer than 3 seconds to launch, a message must be displayed. This may be of the form of a progress dialog, or a “Loading, please wait...” dialog. The application must give some visible indication to the user that the application is not frozen.

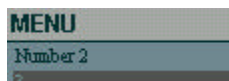
6.5.2

If a splash screen is used in the application, this screen must only display on initial launch of the application. The splash screen must not display while the application is in use. The splash screen must display for no more than 3 seconds. The splash screen should not contain a large amount of text. If the splash screen contains text, the splash screen must remain long enough for the user to read the text.

6.5.3

The application must launch to a consistent screen each time the application is launched as a “new” instance. The application must not open to different dialogs at different times. Where the application is resumed from a suspended state, the application must return to the same state the application was in prior to being suspended.

6.6 Title Bar



The title bar is a narrow band at the top of the screen indicating the application name, or the context of the current screen.

Figure 9 - Title Bar

6.6.1

The main entry screen must display text appropriate to the application name. e.g. if the application name is “AppDemo”, placing the text “Application Demo” would be appropriate. This text can be displayed in the title bar.

6.6.2

The title bar should appear throughout the application, where possible. Where used, the title should be of the same size, use the same font, and be in the same screen location throughout the application.

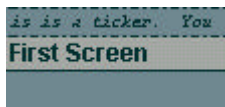
6.6.3

The title bar may also be used to indicate the current screen context. e.g. an application that allows editing of records may have a list view. When the application is in the list view mode, this may be indicated on the title bar. When the application is in edit mode, this may also be indicated in the title bar.

6.6.4

The application name or title must be present upon initial launch of a “new” instance of the application.

6.7 Tickers



A ticker is a scrolling message that appears at the top of the screen above the title bar. A ticker could be used to give information such as headlines, or stock prices.

Figure 10 – Tickers

6.7.1

Information in a ticker should be concise and short.

6.8 Main Menu



The Main Menu is a screen that presents the user with a number of options. These options could be user preferences, an exit option or help dialogs.

Figure 11 - Main Menu

6.8.1

Where an application has a main menu, this should be indicated by a menu icon or the text “MENU” in the center bottom of the display. This indicates that the menu soft button can be pressed to access the menu.

6.8.2

Invoking the main menu, should display a clear list of menu contents. The title bar may indicate the main menu is active.

6.8.3

The main menu should contain an option for closing the main menu and returning to the application. This option should be activated using the right soft button.

6.8.4

While the application is still open, the last selected menu item should be highlighted upon re-entering the same menu, where possible. This gives the user easier access to menu

options that may be performed repeatedly. Note that where the user navigates to a different screen containing the same menu, the position of the highlight in the list should be reset.

6.8.5

The application must have a unique build number. Individual applications within a MIDlet suite must contain a separate version number accessible to the user within the application. This can be done either in a splash screen or the about menu. This build number should take the form of [major.minor.bugfix] For example; 1.3.7 would be major version 1, minor release 3, and bug fix 7. Additionally this build number may have a letter following the build number indicating the phase of the build. e.g. 1.3.7b would indicate a beta build. Other letter indicators can be used such as, “a” for Alpha, “b” for Beta, or “d” for development version. This build number must be consistent across multiple releases. Subsequent releases must be allocated a new build number based on the build revision scheme in use.

6.9 Soft Button indicators



Soft button indicators are used to inform the user which of the soft buttons can be used to activate functionality within the application.

Figure 12 - Soft Button Indicators

6.9.1

Soft button indicators should be at the bottom of the display.

6.9.2

There must be no more than 3 soft button indicators. These must correspond to the Menu soft button, the left soft button and the right soft button. The right soft button should always be used where there is a Select or OK command. The left soft button should always be used where there is a No, Back or Cancel command. Please see section 4.4.3 for additional information regarding the soft key mappings.

6.9.3

The soft button indicators may be surrounded by a border, to make the indicators clearly visible. Where a border is used, the border thickness should be consistent throughout the application. The font size of the soft button indicator should be consistent throughout the application. Where screen size is limited, the borders may be omitted. If a border is not used, the buttons should be clearly distinguished from the remainder of the screen, by a significant space, or a horizontal line separator.

6.9.4

The soft button indicators should be short and should not overlap into the areas used to indicate other soft buttons. e.g. the left soft button indicator should not overlap into the menu soft button indicator or right soft button indicator area. Where the position or size of the button overlaps into areas indicating other buttons, the button to be pressed may be misinterpreted and may cause confusion to the user.

6.9.5

The soft button indicators should clearly indicate the function that is activated upon pressing the appropriate button.

6.9.6

The text of the soft button indicators must not truncate other soft button indicators. Where soft button indicators are truncated, the user may not be able to clearly determine the command associated with the appropriate indicator.

6.9.7

If soft button indicators are not used throughout the application, the function of each of the soft buttons should be clearly indicated in the user documentation.

6.9.8

Where borders are used, the border should be a good size to contain the text. Care should be used to avoid truncation of the button border.

6.9.9

Where soft buttons are not accessible or do not function, this should be clearly indicated by removing the soft button indicator.

6.9.10

When a soft button is pressed, the indicator should briefly indicate that the button has been pressed by the indicator becoming darkened, where possible.

6.9.11

The soft button indicators must be displayed in capitalized text throughout the application.

6.10 Data entry fields



Data entry fields offer the user an area where they can input information. This information could be numeric or alphanumeric.

Figure 13 - Data entry fields

6.10.1

Data entry fields must not allow truncation of data.

6.10.2

The fonts and font size must be clear to the naked eye (i.e. 20x20 vision).

6.10.3

Data entry fields should be clearly distinguishable from non-editable data fields. This may be achieved by surrounding the field with a border, or graying out non-editable fields.

6.10.4

Upon entering a data entry field that contains data, the cursor must appear at the end of the field. Upon entering a data entry field that does not contain data, the cursor must appear at the beginning of the field.

6.10.5

If the contents of a data entry field extend beyond the edge of the data entry field, moving the cursor to the left and right edges of the field should scroll the data entry field, to show text that was previously beyond the data entry field. If multiple text lines are allowed, moving the cursor down and up via the navigation “rocker” keys should allow additional lines of data to be displayed.

6.11 Scroll Bars

Scroll bars can be used to indicate to the user that there are more items in a list than can be contained on the screen.

6.11.1

Scroll bars should be moved when the user navigates up or down from the top or bottom of the list respectively.

6.12 Data Listing/Viewing

6.12.1

The current highlighted items should be clearly indicated. This could be by the item being surrounded by a line drawn box, solid shading, or an asterisk beside the list item.

6.12.2

Scrolling should be smooth, and not be truncated.

6.12.3

List items should be grouped in a logical order. e.g. alphanumeric items should be in alphabetical order, numeric items should be listed in numerical order, or value based, listed in ascending or descending order.

6.12.4

The last highlighted list item should be highlighted when re-entering the list view, where possible. This does not apply to a list containing the same values, but accessible from a different screen in the application.

6.13 Notifications

6.13.1

System notifications (e.g. notification of incoming calls, dispatch messages, or mail messages) must be allowed at all times during the operation of the application.

6.13.2

System notifications must not cause corruption of the screen graphics or cause other adverse effects to the screen content.

6.14 Error Messages

6.14.1

Error messages should be clear and concise and explain the reason for the given error. If appropriate, a solution should be offered. Displaying error messages, with just an error number, or with no description should be avoided

7 Contact Information

We welcome your comments, feedback, or questions regarding this document. To contact us, please send an email to motorola@qpqa.com.