Eagle User Experience & UI Controls Guidelines

Table of Contents

GLOSSARY

EAGLE – whole-goods display with CNH UI and aftermarket functionality

XCN 2050 - current aftermarket display (Trimble UI with only aftermarket functionality)

UI – The visual representation of the screen.

UX – User Experience, how the user interacts with the UI.

Workflow – Sequence of steps defining how content is grouped on each page.

Business Logic – Decision making process (ex: taking GPS position to define Guidance path)

Data Structure – how data is saved/stored (Databases)

OS – operating system, interface to the hardware.

Tap – an oriented touch on Eagle display that triggers the default functionality for a given item.

Tab – key mechanism for organizing content that allows multiple groups of widgets to be contained within a single window, using tabs as a navigational tool for switching between the view.

GNSS – Global Navigation Satellite System

GPS – Global Positioning System

Cards – Cards are used in the Menu section. They represent groups of different functions (i.e. Vehicle Setup, Profile Setup) that take the user in the setup windows.

UDW – User Defined Window representing i.e. feedback from the equipment.

LHA – Left Hand Area

Modal window – a modal window is a child window that requires users to interact with it before they can return to operating the parent view, thus preventing the workflow on the main window.

Sliding drawer – any hidden panel that user can extend by taping or sliding its handle.

Drop-down panel – it is a sliding down panel that belongs to the top bar.

Color Palette – the primary and secondary colors used on Eagle Display.

SCOPE

This document covers Eagle Display User Experience Principles and UI Controls to be used by various groups and help them develop consistent UX/UI.

USER EXPERIENCE OVERVIEW

The mission is to establish an efficient, intuitive, and consistent user experience while following highest standards for Eagle Display user interface.

1. Design Principles

Layout Flow

- Users should be able to find the content they are looking for at a glance;

- The layout should be spaced and organized in such a way as to guide user's visual direction efficiently;

- Logically related UI elements should have a clear visual relationship;

- UI elements should be emphasized based on their relative importance;

- UI elements and "white space" should be distributed in such a way as to achieve the UI goals;

- Scrolling should be kept to a minimum.

Less is more

Keep interface simple, easy, and intuitive to use. Give user just enough information and controls for the best comprehension and ability to complete their goals.

Consistency, Familiarity and Intuitiveness

- Industry established and familiar UI/UX patterns should be used to make the application look-and-feel natural and intuitive;

- Using familiar components and UI patterns gives the application an instant sense of familiarity and usability;

- Similar windows or views should use a similar layout;

- UI controls should be used consistently.

Streamlined interactions

- Display interactions should be streamlined to support key use cases;

- Common tasks should be accomplished with the minimum number of steps;

- Layouts should be designed to minimize clutter. The attention should be focused around the key activities and objects, and place less frequent, or less important activities a tap away;

2. Visual Style Objectives

Eagle visual design follows a brand-neutral approach to applying colors and graphics to the UI. There are, however, brand specific representations when it comes to any imagery of a vehicle or implement, which shall be presented in brand specific colors (i.e. icons that incorporate a vehicle or implement).

Following conventions should be followed:

- High contrast and large text increases readability in the cab environment;

- Dimensional elements such as gradients, texture and drop shadows enhance the call-to-action feel for interactive items;

- A primary color (blue) is to be used consistently for call-to-action UI elements;

- Common UI pattern elements should be used when possible. For example arrow shapes indicate expandable content;

- Use icons when possible, visual representations communicate the idea better and faster;

- Following ISO standards for the display UI elements (i.e. icons) helps users associate them with iconography they already know.

3. Standard Gestures to Interact

The Eagle Display utilizes a touchscreen interface. Touch gestures are the primary way of navigating and interacting with the system. The following gestures allow users to directly manipulate content on the screen:

Gesture	Action	Expected result
Touch	Press	Triggers the default functionality for a given item (i.e. initiate an action, activate a control.
Double touch	Two touches in quick succession	Zooms into the map. Also used as a secondary gesture for text selection.
Swipe	Press, move, lift	Scrolls overflowing content; Navigates between pages; Opens and closes sliding drawers.
Long press	Press, wait, lift	Enters data selection mode. (i.e. highlights a sequence of characters, a word, or an UDW.
Drag	Long press, move, lift	Rearranges data within a view, or moves data into a container (e.g. rearranging UDWs, shortcuts or adding tabs into a group of UDWs).
Pinch open	2-finger press, move outwards, lift	Zooms into content.
Pinch close	2-finger press, move inwards, lift	Zooms out of content.

Eagle User Experience

Eagle User Experience framework has been designed to consistently display navigational items and provide user customizable areas for equipment controls and feedback.

Eagle Display is divided into several distinct screen regions:

- Top Bar (1);

  Provide access to global level information, tools, and setups

- Left Hand Area (2);

  Contains vehicle performance and precision farming information.

- Runscreen (3)

  Contains controls that users need daily in order to efficiently operate the equipment and productively accomplish the job.

- Bottom Bar (4);

  Contains runscreen page controls, status and shortcuts drawers.

Figure 1. *Eagle Display Regions*

The regions help organize the data better and provide the user with a system for understanding where certain items, controls, or interactions are available.

The content of the runscreen and bottom panel of the LHA can be customized by the user in Edit Screen Layout Mode (for more see Edit Screen Layout Mode/Edit LHA bottom panel, page 35).

1. Top Bar

The top bar represents the upmost, the full width region which is delimited by the gray background. The top bar stays always on the screen and must not be covered by pop-ups or other type of UI overlays. The goal of the Top Bar region is to provide access to global level information, tools, and setups, which include:

1.   Alerts Tray
2.   Help
3.   GNSS Status
4.   Instant Messages
5.   Weather
6.   Clock
7.   Menu
8.   Home
9.   Login
10. Auto Engage Button

Figure 1.1. *Top Bar*

The Clock shows the time and divides the top bar content as well. The clock's position is fixed. The amount of items on both sides of it may differ depending on the available equipment and services.
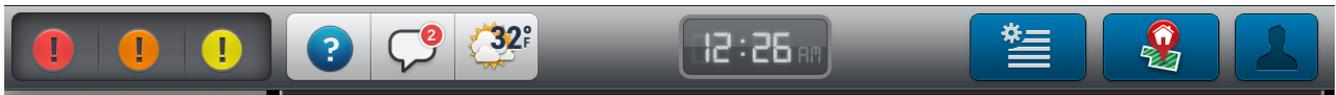
Figure 1.2. *Top Bar, no Auto-Engage button*

If any of the buttons to the left or right are absent in a given setup, the remaining elements will be aligned to the left or right correspondingly (i.e. if auto-guidance system is not available then, the far right auto-engage button will not show up in the top bar and the blue buttons to the right of the clock will be shifted and aligned to the right margin.)

1.1. Alerts Tray

Alerts appear in the leftmost portion of the top bar. The alert icons on the button represent the 3 levels of alert severity in the system: high, medium, and low (from left to right).

Figure 1.3. *Alert Sections in Top Bar*

An additional number indicator above the icon shows the total number of currently active alerts for that specific severity level.

Pressing anywhere within the alert area will open a drop-down containing the Alert History over the runscreen area. To close the drop-down, the user can press anywhere outside of the panel or press one more time on the alert button in the top bar.
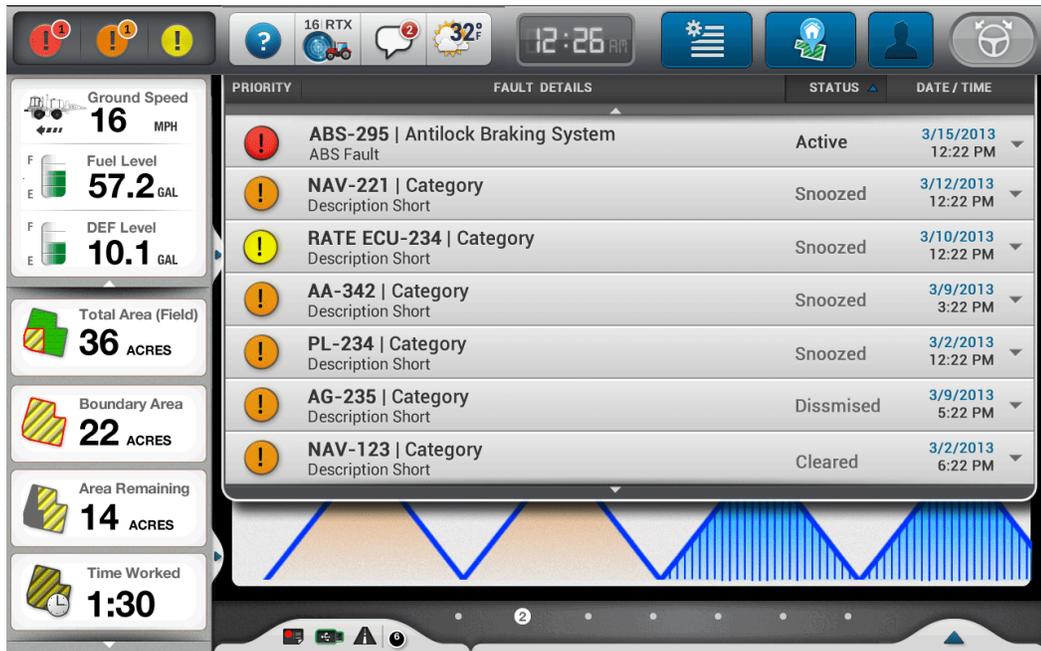
Figure 1.4. *Alerts History Panel*

When there are no current alerts, nor alerts in the history stack, pressing on the alerts button will still open the drop-down which will show a message informing the user that there are no Registered Faults.
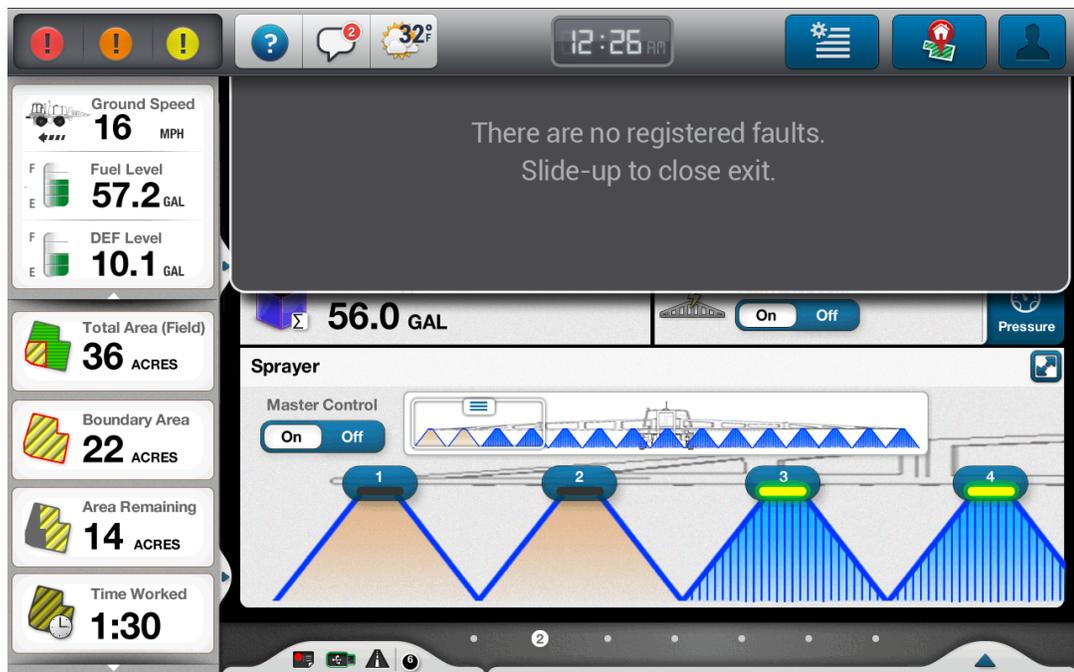
Figure 1.5. *Alerts History Panel, empty stack*

1.2 Help Button

The Help button opens drop-down which is meant to provide users with assistance and instruction about the system as well as customer support contact info.

Figure 1.6. *Help, Manuals Tab*
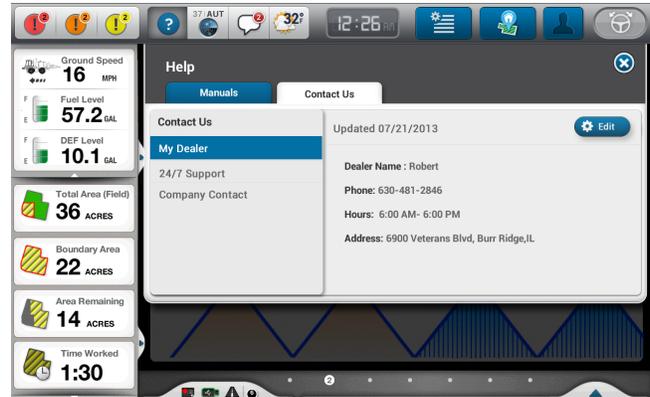
Figure 1.7. *Help, Contact Us Tab*

1.3. GNSS Status Button

The GNSS button displays the current status of the guidance system, which can be setup or adjusted from the Menu (see). If both Vehicle and Implement are equipped with GPS receivers, the vehicle and implement signal status will alternate on the button every 3 seconds. Users have the ability to lock/unlock one or the other from the GNSS drop-down panel, which slides down when user presses on the GNSS button.

The GNSS drop-down contains more detailed, read-only information. The user can choose to view the Vehicle or Implement information by pressing on the specific tab. Once on the Vehicle or Implement tab, user can view detailed information related to Status, Skyplot or Position.

Figure 1.8. *GNSS drop-down, Skyplot*

Figure 1.9. *GNSS drop-down, Status*

1.4. Instant Messaging Button

The button contains a message icon along with a number indicator that shows the amount of unread messages. On press, the drop-down slides out. Here, the user can receive and read messages sent by others as well as reply.
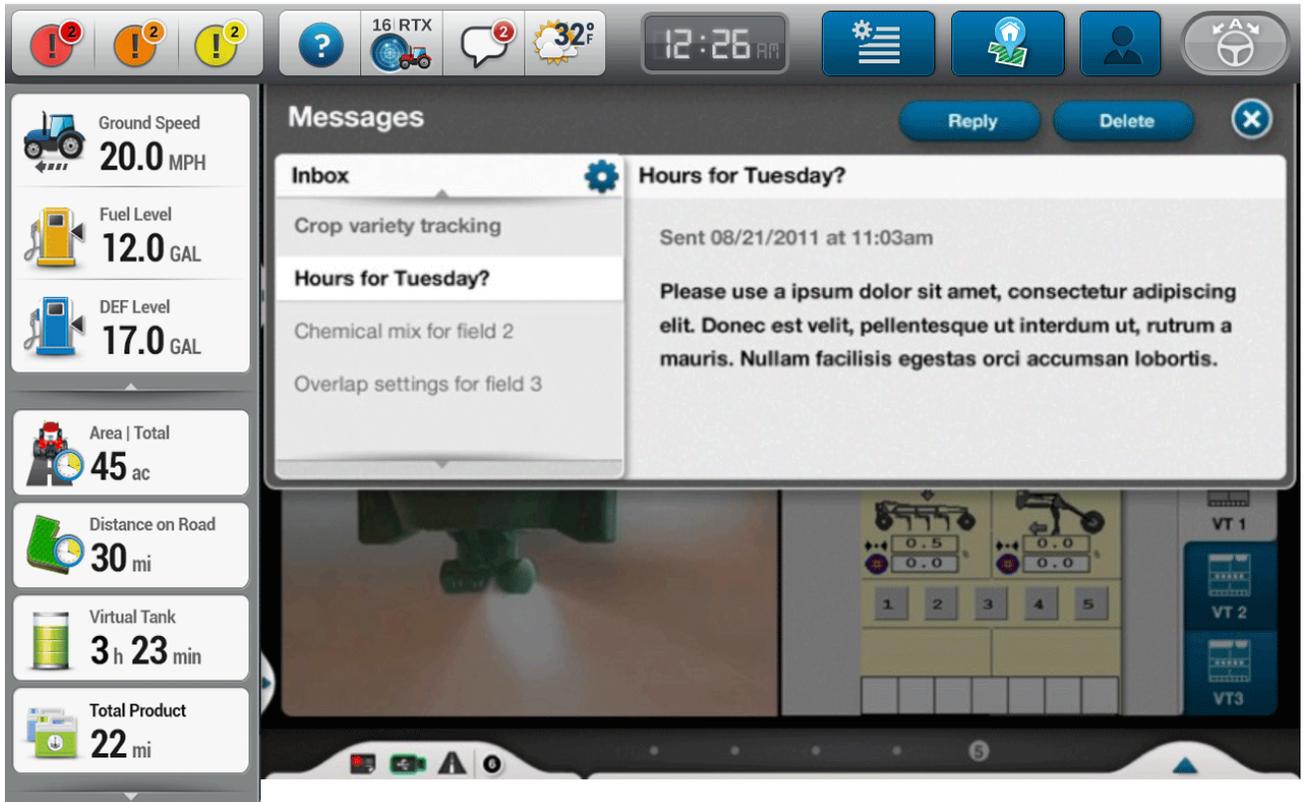
Figure 1.10. *Instant Messaging Drop-Down*

1.5. Weather

The Weather button provides real-time weather information. The icon updates live to display the current outside temperature and conditions in the top bar. On touch, the Weather drop-down opens. Here, the user can view the current and the immediate future weather temperature and conditions, the weather map as well as access weather settings.

Figure 1.11. *Weather Sliding Panel*

1.6 Menu Button

On press the general menu pop-up window shows up. It contains all system settings, diagnostics, and productivity reports organized on separate tabs, which allow users to switch between them.

Figure 1.12. *Menu, Menu Cards*

Within each tab, the content is logically grouped into multiple Cards (i.e. System, User Profile, Implement Setup etc.). Pressing a card will take the user one level deeper into the hierarchy to a detail page for the group.

Figure 1.13. *Detail Page*

Within the detail pages the information and controls are logically organized into vertical tabs, allowing users to switch between topics within that Card. Each Vertical tab has a unique name and icon representing the grouped content. When the Name of the Group is too long a shorter version of it should be defined in the requirements in order to fit in the tab.

The main area of the detail page becomes scrollable if the content does not fit into one single view.

If the card contains more than 5 vertical tabs, then a visual scroll indicator should show up and the user should be able to scroll the tabs.

Best Practices:

Having two big vertical scrollable areas so close to each other is not recommended. If you have more than 5 tabs you should start thinking about splitting this card into two if possible.

1.7 Home

Pressing the Home button in the top bar will open the Home pop-up window, which contains 3 sections: Field information, Vehicle and Implement Information and Product information.

This window is designed to be the place where the user would go first before starting a job. Here users can manage and filter field related data, select vehicle, implement and their configuration as well as other job related setups.

The geo-location map is also available, by pressing on the locate button on the map, the system will try to locate the field if such has been recorded previously.
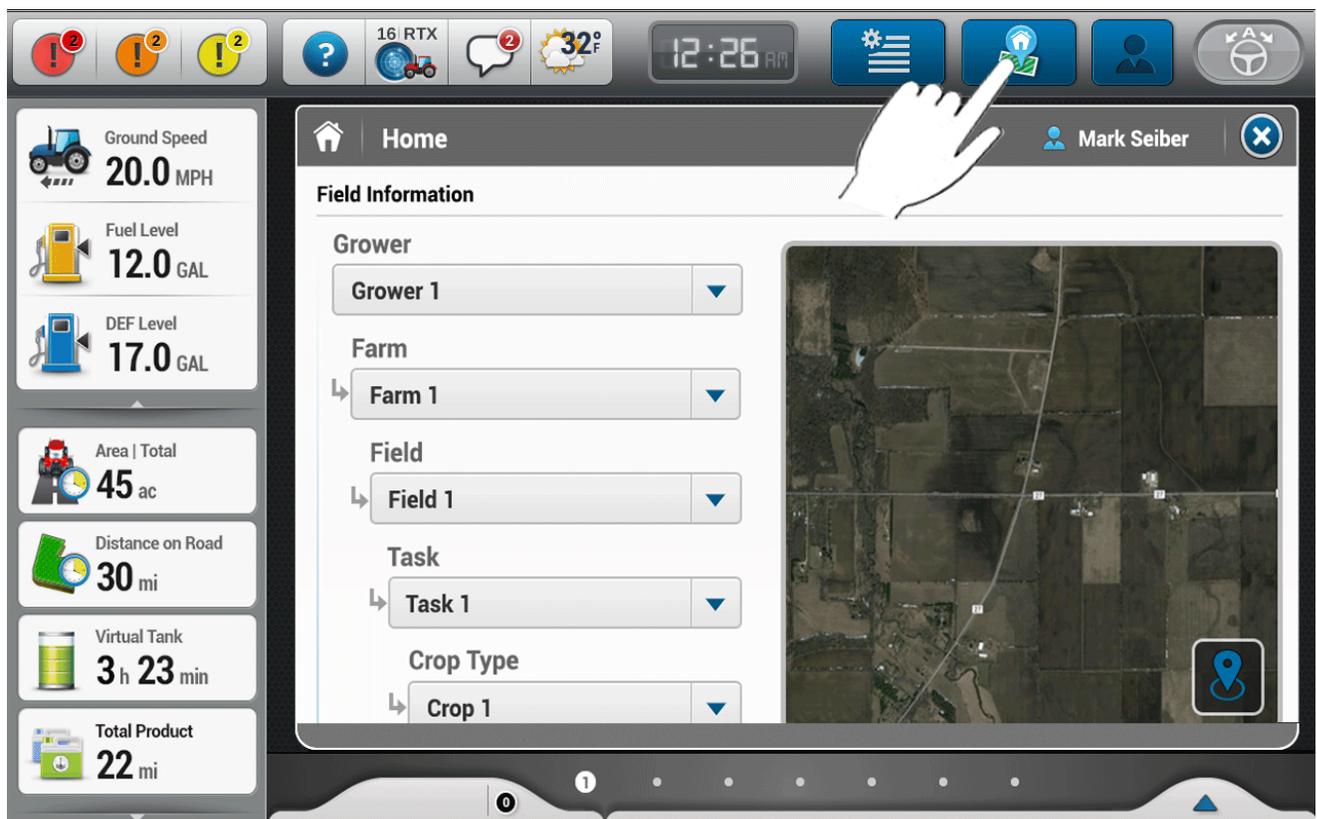
Figure 1.14. *Home Window*

Figure 1.15. *Vehicle Information in the Home Window*

Figure 1.16. *Product Assignment in the Home Window*

1.8. Login Button

On tap, the login button brings up a pop-up that allows user to login to their account. Restore password functionality is also available.
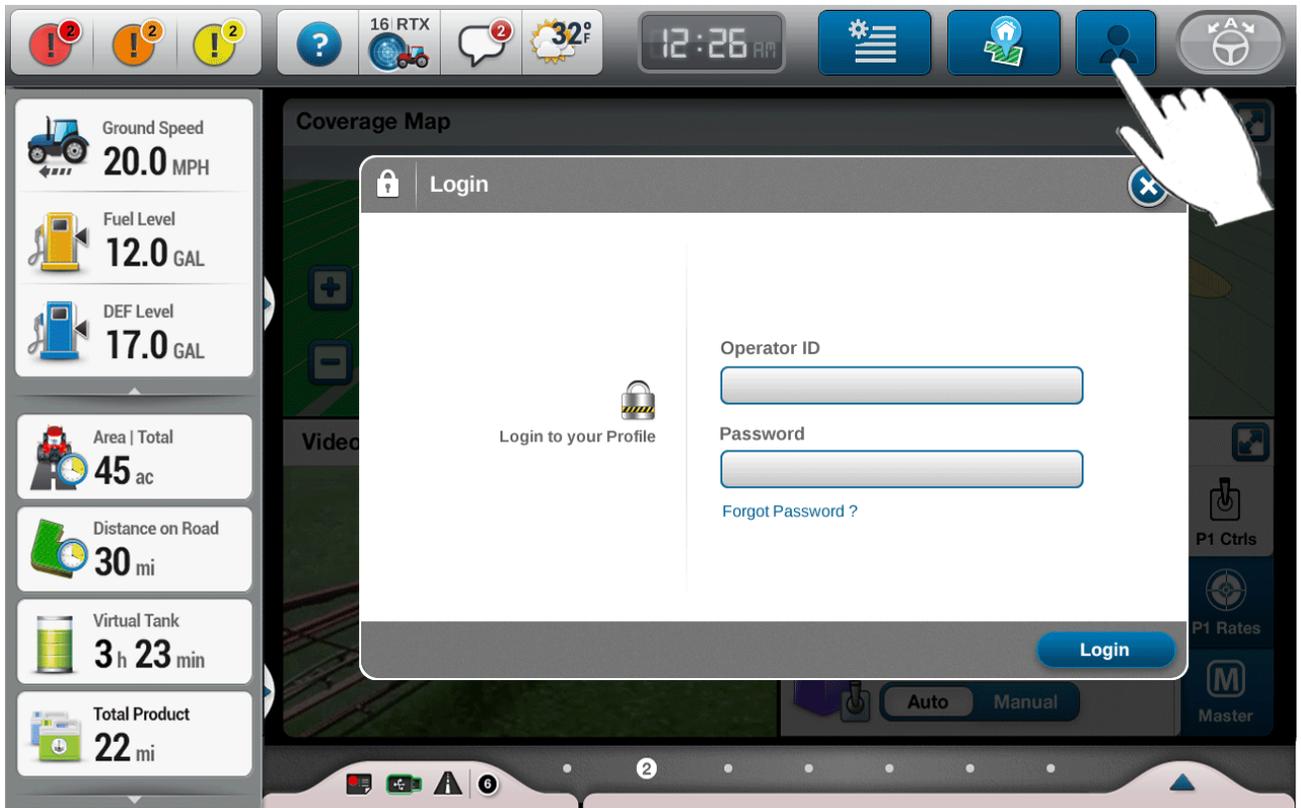
Figure 1.17. *Login Window*

1.9. Auto-Engage Button

Auto-engage button will show up only if Auto-guidance system is available for this particular vehicle. The auto-engage button has multiple states and it works similar to a switch between its states.

The placement of the button in the rightmost allows the user to easily press it while gripping the right side of the display.

Figure 1.18. *Auto Guidance Engage/Disengage*

2. Left Hand Area (LHA)

LHA stays always on the screen and must not be covered by pop-ups or other type of UI overlays. The left hand area is organized in two sections (can also have three sections, if required by the platform):

- Left Hand Area Top

    The fixed size top part, which contains vehicle information.

- Left Hand Area Bottom

    The scrollable bottom part, which contains precision farming related UDWs.

Figure 2. *Left Hand Area*

2.1. LHA Top Panel

The top section of the LHA is dedicated for vehicle information.

The amount and type of information displayed, depends on the vehicle platform. The LHA top panel can accommodate any amount of read-only values between two to five.

By swiping horizontally (from left to right) anywhere within LHA Top panel, or by dragging the handle coming from below it to the right, users can open the full LHA Top drawer that displays additional vehicle information, which is also customizable by the vehicle platform. The LHA Top Drawer covers the whole runscreen.

Figure 2.1. *Expanded LHA Top Drawer*

User can swipe from right to left or tap and drag the handle to the left to close the panel.

2.2. LHA Bottom Panel

The LHA Bottom Panel's height covers all the area that is left under the LHA Top panel but it should be not less then enough to accommodated a minimum of two UDWs. This area is scrollable, visual scroll indicators are present.

LHA Bottom comes with a default set of precision farming and vehicle controls but the panel is ultimately customizable by the user. Users can delete, rearrange or add UDWs to the LHA Bottom panel at any time, though the content can be edited only when the panel is closed. (for more see)

Expanding the Panel

By swiping horizontally (from left to right) anywhere within the scrolling area, or by dragging the handle coming from below it to the right, users can open the LHA Bottom drawer that displays the whole list of UDWs.

The LHA Bottom Drawer covers the bottom part of Run Screen area, its height being defined by the height of the LHA Bottom panel. The Run Screen dims slightly, but continues to update and provide live information. The shortcuts and status drawers are still available. Users can close the panel by swiping horizontally from right to left, by dragging the handle to the left or by taping anywhere on the dimmed runscreen.

Figure 2.2. Expanded LHA Bottom Drawer

Figure 2.3. *Expanded LHA Bottom Drawer UDW (vertical)*

3. Runscreen

The runscreen is the centerpiece of the display. Its goal is to facilitate the direct access to the controls that users need daily in order to efficiently operate the equipment and productively accomplish the job.

Figure 3. *Runscreen Window*

The runscreen can have up to 7 pages. Users can navigate through them by swiping through runscreen area or by taping on the page controls in the bottom bar.

The runscreen pages will come with default page configurations for each type of vehicle, which will be defined by the vehicle platform. In the same time the runscreen is highly customizable.

The runscreen base elements are the user defined window (UDW) and groups of UDWs either predefined by the system or created by the user.

3.1 Runscreen Layout Principles

The Runscreen Layout follows the rules:

- The whole runscreen is divided into a grid of 12 cells (2 x 6);

- The size of a single cell is designed to accommodate the smallest size UDW (1x1);

- Multiple UDW's can exist separately as well as in a group, a group covers a minimum of three cells;

- Single UDWs do not have a title;

- A group together with the group title covers exactly the same number of cells as the UDWs that form the group would cover together if arranged vertically one after another.

- The groups can be organized in tabs

See examples of possible runscreen layouts below:
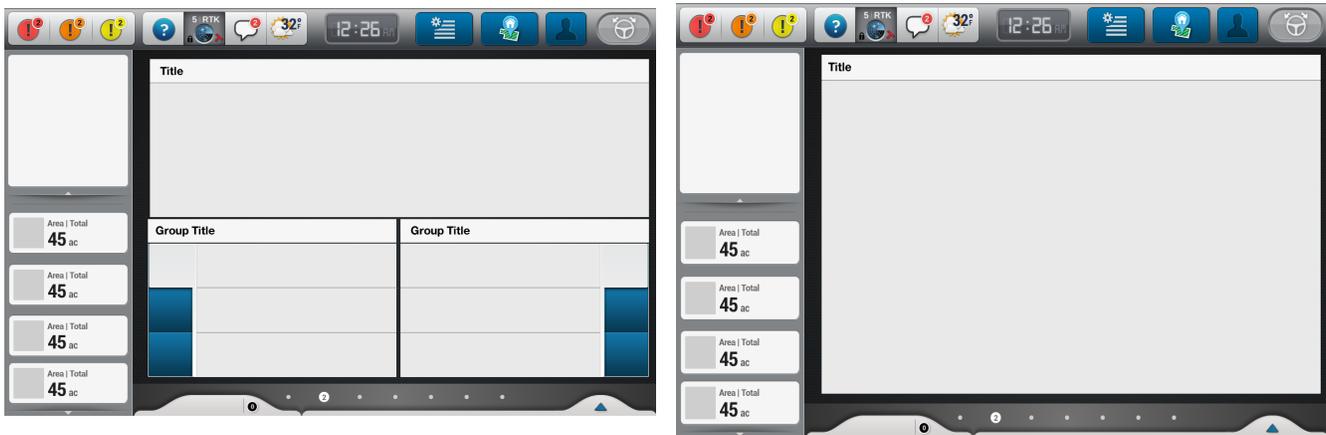
Figure 3.1. *Example of Runscreen Layouts*

3.2. UDW

- UDW represents only one piece of information or control (i.e. total area, nudge or swath recorder).

- The smallest UDW can cover only one cell (1x1) while the biggest can cover the whole runscreen area thus all twelve cells (2x6).

- The UDW's can have different sizes, which have to snap to the cell grid (i.e. 1x1, 1x3, 2x3 etc.).

Examples:
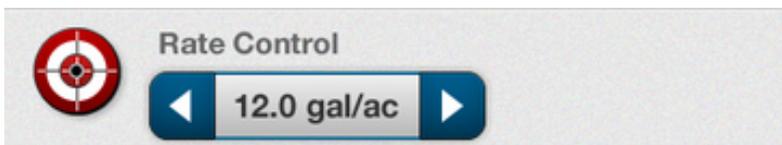
Figure 3.2. *Smallest UDW, 1 cell (2x1)*

Figure 3.3. *Smallest UDW, 1 cell (2x1)*

Figure 3.4. *Quarter UDW, covers 3 cells (1x3)*
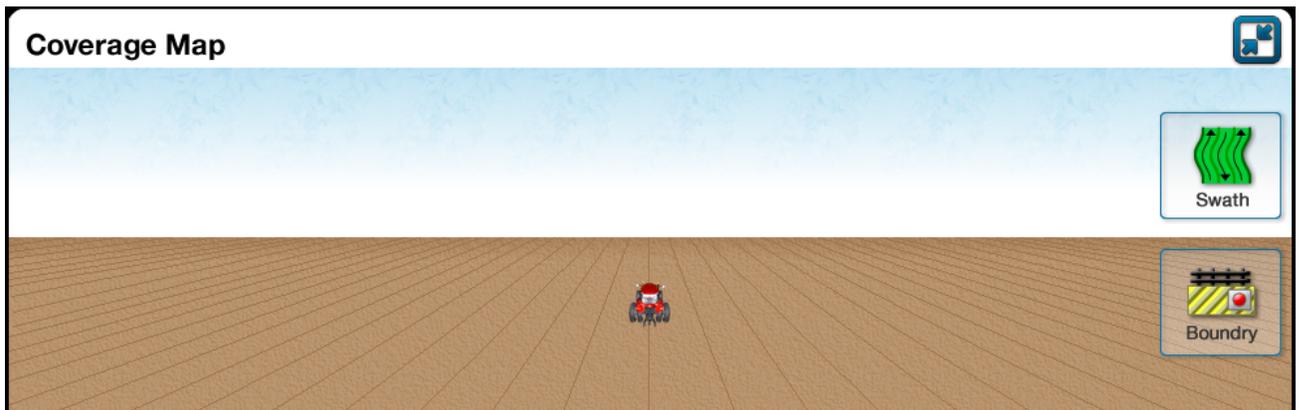
Figure 3.5. *Half page UDW, covers 6 cells, (2x3)*
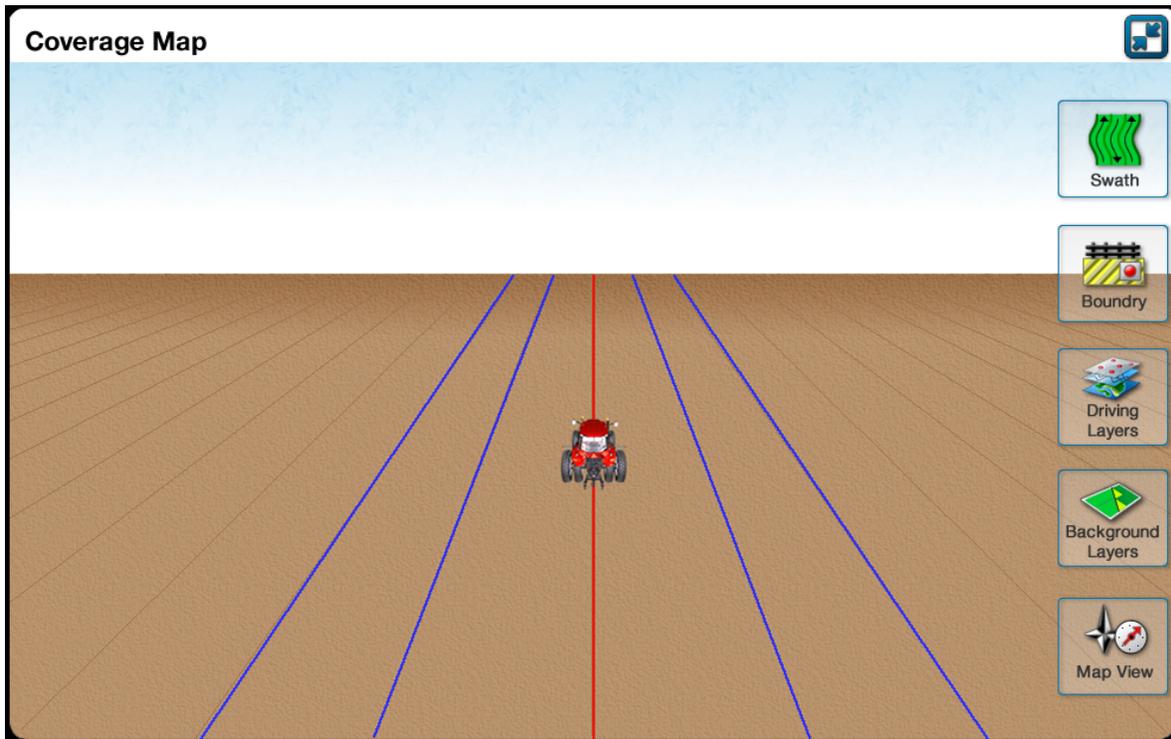
Figure 3.6. *Full page UDW, covers 12 cells, (2x6)*

3.3. Groups of UDWs

The groups of UDWs are either predefined by the system or created by the user. The predefined groups are different for each vehicle and implement type and are defined by the vehicle platform. The user-defined groups of UDWs can be created in Edit Screen Layout Mode *(for more see Edit Screen Layout Mode/Edit Runscreen, page 34).*
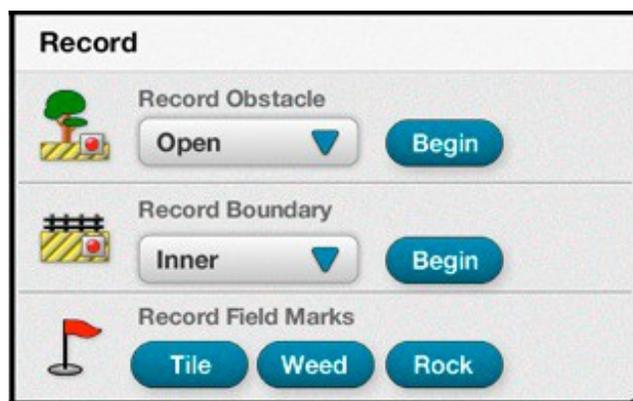
Figure 3.7. *A group of 3 UDWs, covers 3 cells of the grid*
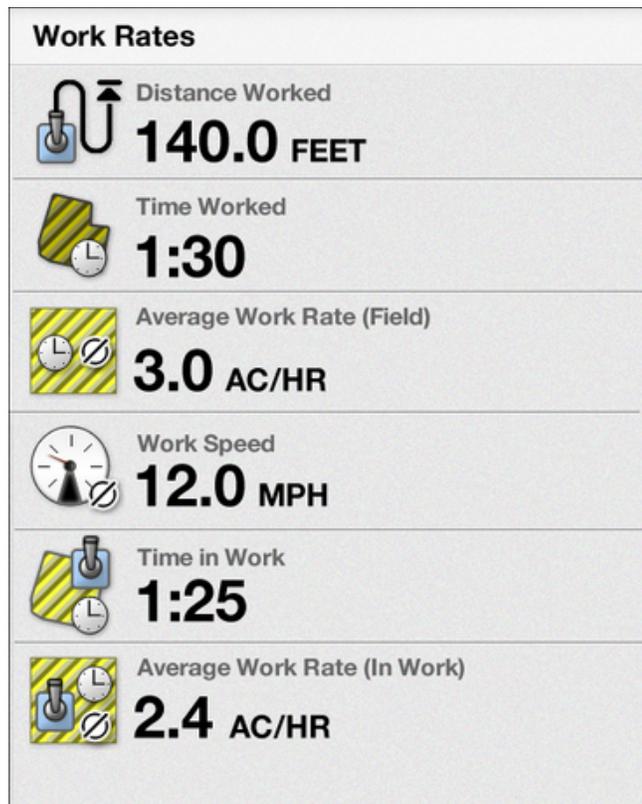
Figure 3.8. *A group of 6 UDWs, covers 6 cells of the grid*

The groups of UDWs can be organized in tabs (for <mark>more</mark> see).

Figure 3.9. *Groups organized in tabs*

4. Bottom Bar

This area contains:

- Status drawer (1);

- Runscreen page controls (2);

- Shortcuts drawer (3).
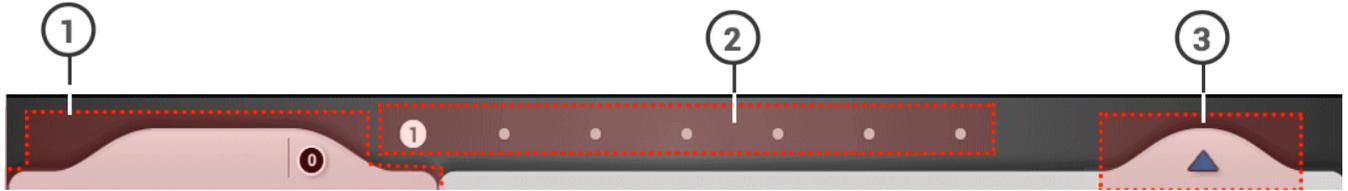
Figure 4. Bottom Bar

4.1. Status Drawer

The status drawer displays current, read-only statuses (e.g. system, vehicle, equipment, task in progress, etc.). The drawer handle shows the three newest incoming statuses and an additional number indicator that shows the number of current status notifications. The open drawer displays all the statuses and their short description and it is scrollable if needed.

To open the drawer, users must tap on the handle or slide their finger from bottom to top. The drawer slides up on top of the Run Screen, which is dimmed while the data is still updating live, in the background. To close it, users can touch anywhere outside of the panel, tap on the handle or slide the handled from top to down.

Figure 4.1. *Status Drawer*

4.2. Run Screen Pagination

The run screen page count and a visual indicator of the current page shall be displayed on the bottom bar.

Figure 4.2. *Runscreen Pagination*

4.3. Shortcuts Drawer

This drawer contains the quick access shortcuts that the user added previously by starring certain setups; productivity or diagnostics detail pages from under the Menu.

Pressing a shortcut button will take the user to the appropriate Menu screen.

If the user did not add any shortcuts yet, an instructional message of how to add a shortcut will be displayed.

Shortcuts can be edited by pressing the icon in the upper-right corner of the drawer (i.e. delete and rearrange by drag and drop).

In addition, this drawer contains the Edit Screen Layout button, which will take the user into the Edit Screen Mode. When in Edit Screen Mode, the bottom drawer will be replaced with Edit Mode Controls (i.e. add, delete, restore defaults etc.).

To open the drawer, users must tap on the handle or slide their finger from bottom to top. The drawer slides up to open on top of the Run Screen, which is dimmed while the data is still updating live, in the background. To close it, users can touch anywhere outside of the panel, tap on the handle or slide the handle from top to down.

Figure 4.3. *Shortcuts Sliding Panel*

5. Edit Screen Layout Mode

To enter the Edit Screen Layout mode user should tap on the Edit Screen Layout button from the shortcuts drawer. A handle icon will show up for the items that can be edited. To edit any item, which is already placed on the screen, the user has to long press in order to select it first.

5. 1. Edit Runscreen Pages

Unlike in normal view where the page number is displayed only for current runscreen page in Edit Mode all the pages are numbered. Each page representation has an additional visual indicator showing which page has content and which is empty. The handle icon below the page shows the user that they can long press to drag-and-drop if they wish to rearrange the pages.
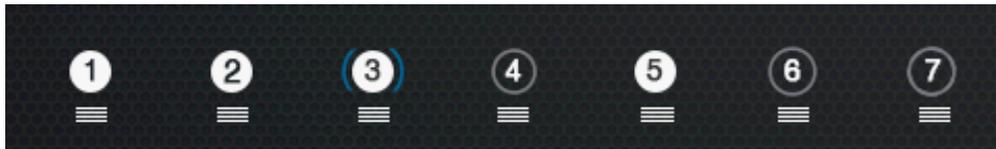
Figure 5. *Runscreen Pagination in Edit Mode*

When exiting the Edit Mode the pages shall be automatically rearranged and renumbered so that the empty pages are moved to the end.

In regular mode only the pages that have content shall be displayed.

If for any reason, user deleted the content from all the pages, the first page will show as selected and the runscreen will appear empty.

When entering the edit mode, the user shall be set to the current page they were on before entering the Edit Mode.

5. 2. Edit Runscreen

Figure 5.2. *Edit Mode*

User can group and rearrange UDWs on a particular runscreen page by dragging them onto the desired available area.

The UDWs can be organized in groups by dragging them one over another, while groups can be organized together in tabs by dragging an item over the tab designated area.

To delete a UDW or a group of UDWs user must select and drag them over the delete icon or select and tap on delete.

To add an UDW or a group of UDWs, users must press on the add button which will open the UDWs Browser.
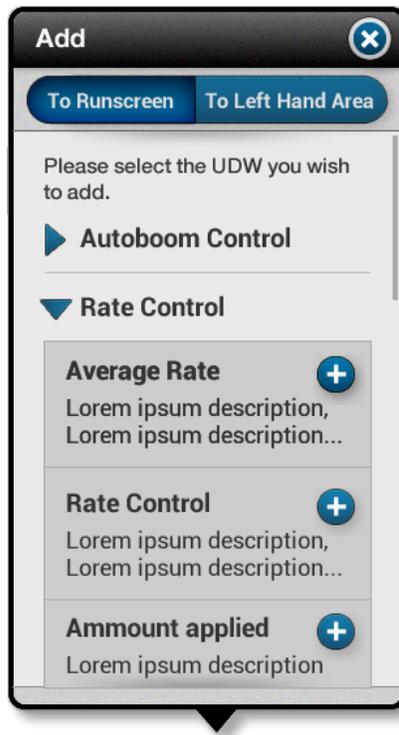
Figure 5.3. *UDW browser*

When trying to select UDW's or UDW groups from the UDW browser to be added to the runscreen, the ones that would not fit in any configuration of the current runscreen page, will be shown visually in a disabled state and will not be accessible.

When user starts dragging UDW or a group out of UDW browser, the UDW browser closes. If user decides that they do not want to add that particular UDW or the group of UDWs anywhere on the runscreen, user can drag over the delete icon.

5.3. Edit LHA Bottom panel

Adding or deleting UDWs as well as rearranging them can customize the list of UDWs in the LHA bottom panel. It is important to mention that the list above cannot be customized when the LHA bottom drawer is open.

To add a UDW, user must tap on the plus button and select to add to the LHA, select the desired UDW and drag it onto the panel over the area where it should be placed, at this moment the list items will move up and down in order to accommodate the UDW to be added.

To delete a UDW, user must select and drag them over the delete icon or select and tap on delete.

To rearrange a UDW, user must tap on it to select, then, move over the new area where UDW will be placed. While dragging the user can hover over the scroll control in order to scroll the list of UDWs in search for the selected UDW new placement.

6. Pop-up Windows

Pop-up windows open on top of runscreen, and may be of any size within the constraints of the runscreen area. These popups are typically modal, which forces the user to interact, acknowledge, close, or dismiss the window before returning to the Run Screens (e.g. Menu, Home).

When a window has been opened, the runscreen is dimmed. The content of the runscreen will continue to update, but will not be interactive.

Figure 6. *Menu*

Figure 6.1. *Home*

7. Alert windows

The alerts inform users of a problem that has already occurred. Effective alerts inform users that a problem occurred, explain why it happened, and provide instructions about how it can be fixed.

The Alerts are categorized by their priority as high, medium and low priority alerts.

The Alert windows are black, touch-through, semi-transparent windows, which show above top-right corner of the Runscreen when an alert of any severity comes in. The window has a stroke around it that matches the fault severity. The alert windows animate in from right to left.

Figure 7. *High Priority Alert*

Figure 7.1. *Medium Priority Alert*

Figure 7.2. *Low Priority Alert*

8. Slap Windows

This window is designed to show immediate feedback as a result of the user adjusting physical controls on the vehicle dashboard.

Figure 8. *Example of a Slap Window with a Dynamic Graphic*

- Will show on the top right of the run screen area, menus, and detail pages, without closing them.

- Shall appear on the screen immediately as users start adjusting parameters from physical controls.

- Shall stay on the screen for the whole process of adjusting values from the physical controls.

- Shall stay on the screen for another three seconds after the user stops adjusting the physical control

- If the user tap on the screen, the slap window shall disappear, assuming users are trying to get directly to the display's functionality, if the user starts adjusting the physical controls again then the slap window will return.

- Slap window represents only one adjustment at a time. If the user is adjusting multiple parameters, then slap window stays on the display, and its content changes to exactly the parameter that is being adjusted at that same millisecond.

- The slap window will be displayed despite the fact that the parameter being adjusted is already shown on the active run screen or active detail page, as it is representing the feedback to the users' adjustment from the physical controls.

- It is a read only window.

9. Dialog Boxes

A typical dialog box appears as an overlay window on top of the system content area and it is centered on screen. It consists of a title bar (to identify the command, feature, or program where a dialog box came from), an optional main instruction (to explain the user's objective with the dialog box), various controls (to present options), and commit buttons.

It is usually used to:

- allow users to perform a command;

- gets input from the user;

- provides users with information or progress feedback.

Dialog boxes, being modal elements, are disruptive to the user's workflow, and therefore their usage should be minimized as much as possible.

Use Case Examples:

- **Query Dialog:** Used to verify that the user wants to proceed with a destructive or unrecoverable action, for example deleting a file.

- **Information Dialog:** Displays important information to the user and requires a response.

- **Confirmation dialog:** User is prompted with questions or information summary dialog box.

- **Selection Dialog:** Enables the user to select an item or value from a set, based on a basic dialog component that can contain various UI elements such as lists etc.

- **Input Dialog:** Provides a way to input short text or character strings. Contains one or more input fields, together with confirming and cancelling actions. The confirm action can be disabled until all required fields are completed.

- **Machine Initiated:** used when a system event/action requires input from the user.

- **Progress Dialog:** Informs the user of the status of a particular process. Contains text describing the active progress and the application it relates to. Illustrates the status of the process and has an action that enables the user to terminate the process. The dialog box closes automatically when the process is completed. (*see Progress Indicators).*
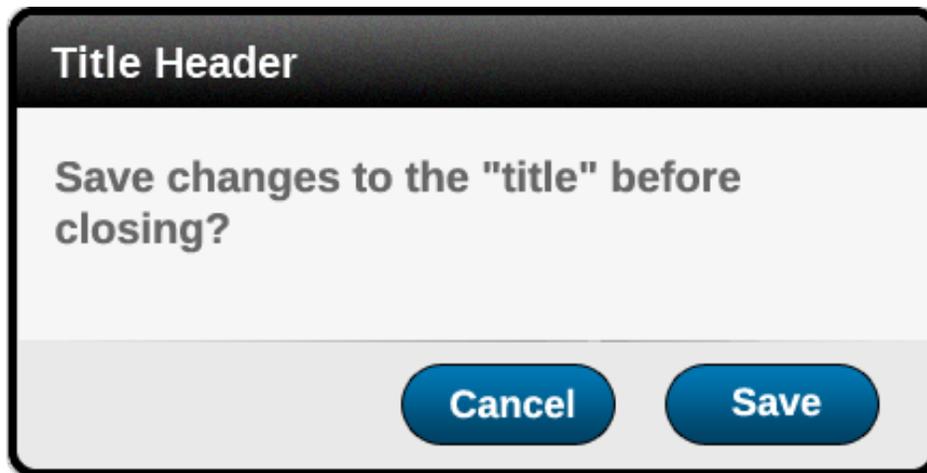
Figure 9. *Confirmation Dialog.*

Dialog Box Layouts

The number of commit buttons should be limited to a maximum of four.

- One button layout (usually used to convey a message):

  o One Close or Cancel Button.

  o The goal is to make sure that the user had time to read the text, as the text might be long, and/or might appear with a delay from the event that triggered it during which user's attention might be lost from the screen.

- Two buttons layout (usually used for confirmation query):

  o Two Buttons placed side-by-side.

  o The right button is the positive choice for the task flow; (e.g. "Continue your flow? No/ Yes "; Yes being on the right.)

  o The left choice is to stop the current task; (e.g. "Delete items? No/ Yes "; No being at the left.)

10. Eagle UI Interactions Priority Specification

There shall be an order of priority given to the display screen interactions. These rules allow the prediction of which controls and widgets will be available to the user to interface with, in the event when multiple UI/UX layers are displayed on the screen at the same time. Each 'layer' shall have a priority. Those layers with higher priority shall overlay those with lower priority. If lower priority items are being initiated while a higher priority layer is on top, its content shall appear below and the user shall be able to interact with it once all the higher priority content is cleared. The list below shall be the hierarchal order interaction as it appears to the user, with 1 being the highest priority layer and the subsequent layers having reduced priority:

1. **Slap windows (system generated)**

   Slap windows have high priority and will always stay on the top of any other window. Slap windows always appear on the left side of the run screen. The window shall stay for three or five seconds (based on the requirements).

2. **High Priority Fault; (system generated)**

   High Priority faults shall slide out from the right side of Eagle Display, below the top bar.

3. **Medium Priority Fault (system generated)**

   Medium fault shall slide out from the right side of Eagle Display, below the top bar.

4. **Low Priority Fault (system generated)**

   Low Priority Fault shall slide out from the right side of Eagle Display, below the top bar.

5. **Dialog Boxes**

   Dialog boxes appear will show up centered over the top half of the run screen

6. **The Top Bar drop downs, LHA drawers and Bottom Drawers**

7. **Menu, Field and Login Screens**

8. **Run Screens**

Only one drop down or sliding drawer (i.e. faults, gps, help, LHA drawers) can stay open at a time. Opening another drop down or sliding drawer will close the first one. If any of the drop down or sliding drawers are opened when the user initiates a popup window such as Menu, Field or Login, the opened drop down or drawer will close.

If a system generated dialog window was initiated while a drop down or drawer was opened, it will appear on top of the drawer or panel. After the window is cleared the user shall get back to the same state of the display as they were before.

COLOR PALETTE

Eagle color palette is divided in two categories: primary and secondary colors.

Each color has lighter or darker shades that can be used as a complement when needed.

Primary Color Palette

The primary color palette is defined to be used consistently for all the widgets, controls, windows and backgrounds.

Blue and its shades are the standard accent colors, used for call-to-action UI elements or when emphasis is needed, while gray with its shades is the passive color used for all the other UI components.

Color	Name	Value
	E Blue Dark	RGB 57/39/91 HEX #39275b
	E Black	RGB 0/0/0 HEX #000000
	E Grey Dark	RGB 68/68/68 HEX #444444
	E Grey Medium	RGB 185/185/185 HEX # b9b9b9
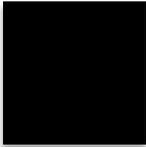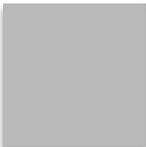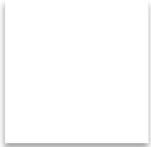
Color	Name	Value
	E White	RGB 255/255/255 HEX # ffffff

Secondary Color Palette

The secondary colors are used mainly for emphasis. Choose colors that fit with eagles brand and provide good contrast between visual components.

Color	Name	Value
	E Blue Medium	RGB 0/120/179 HEX #0078b3e
	E Blue Light	RGB 92/148/167 HEX #5c94a7
	E Red	RGB 231/67/65 HEX #e74341
	E Yellow	RGB 218/211/12 HEX #dad30c
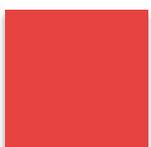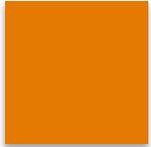
	E Orange	RGB 228/122/1
		HEX #e47a01
	E Grey Light	RGB 238/238/238
		HEX # b9b9b9
	E Cyan Light	RGB 204/221/221
		HEX #ccdddd

Gradients

Gradient	Name	Value
	E Blue Gradient	**Start Color RGB:** 0/120/180
		End Color RGB: 0/65/99
		Start Color HEX: 0078b4
		End Color Hex: 004163

TYPOGRAPHY

Only one font is used through Eagle display, which is Roboto. Bold, Bold condensed or regular styles can be used depending on the following specifications:
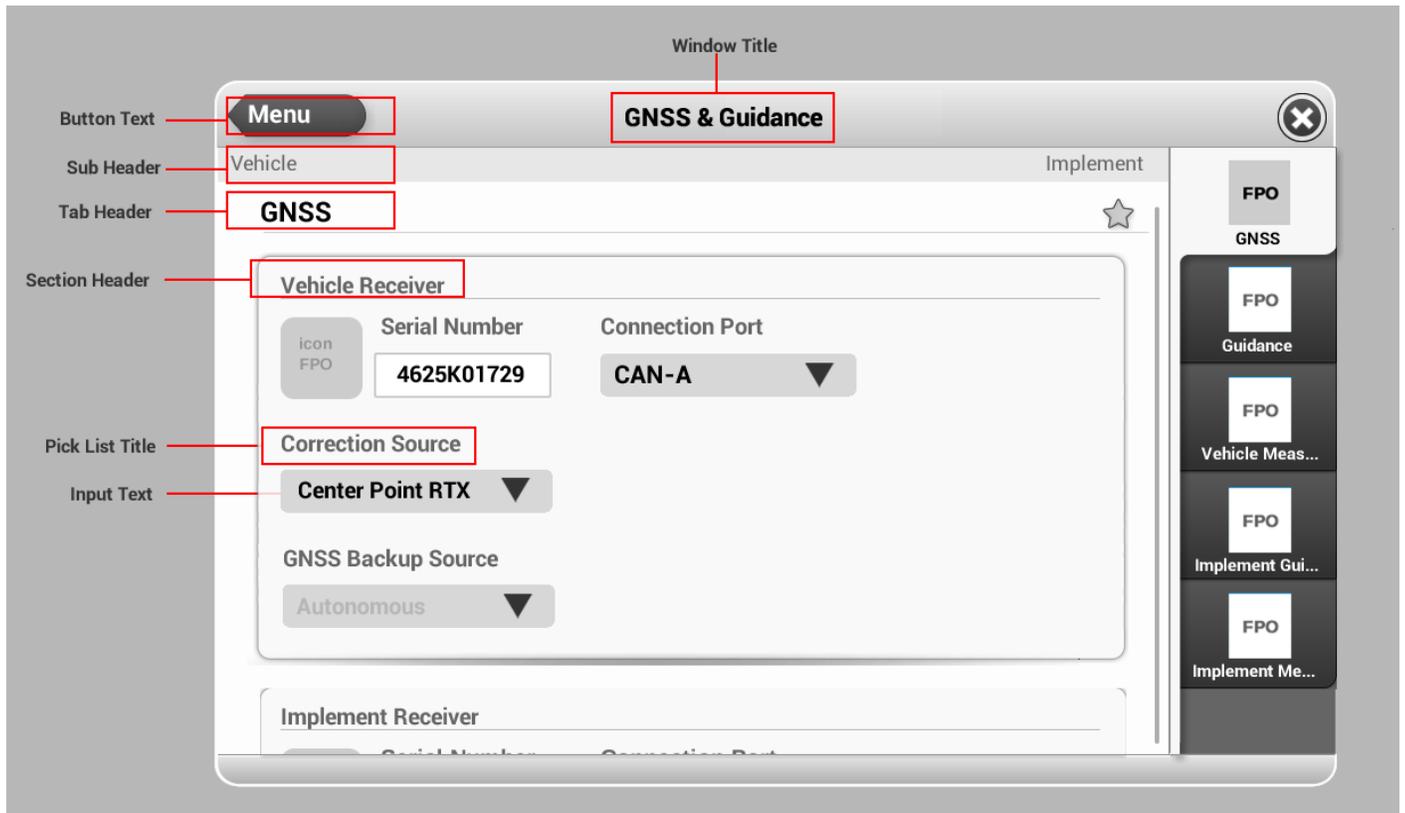
- Window Title: 22px (Roboto/ Bold), Color: E Black

- Button Text: 22px (Roboto/ Bold), Active Color: E White

- Sub header: 18px (Roboto/ Regular), Color: E Grey Medium (#666666)

- Tab Header: 24px (Roboto/ Bold), Color: E Black

- Section Header: 20px (Roboto/ Bold), Color: E Grey Medium (#666666)

- Pick List Title: 20px (Roboto/ Regular), Color: E Grey Medium (#666666)

- Input Text: 20px (Roboto/ Regular), Color: E black

UI CONTROLS

The Eagle application has multiple widgets and controls that can be used to accomplish UX/UI workflows.

1. Touch Target Size

Eagle Display's touch screen facilitates direct manipulation of content and controls, for that the UI call-to-action elements have to be big enough to capture fingertip actions without frustrating users with inaccurate and tiny targets.

The standard size of the tap target is 56 x 56 pixels, which is bigger than the industry standards assuming Eagle Display's cab environment.

The minimum touchable target size is 48 x 48 pixels, which is no smaller than the smallest average finger pad, that is, no smaller than 1 cm (0.4") in diameter.

By defining the square target area does not necessarily mean that what the user sees—the outline of the button itself—has to spread over that entire area.

For example;

- The shortcut icon (star) inside the detail page header bar, for example, is only 30 pixels high, but its tap area extends to the full 56-pixel square.
- The close button at the top right of the detail page header bar has a 48px high and a 56px wide touch area.
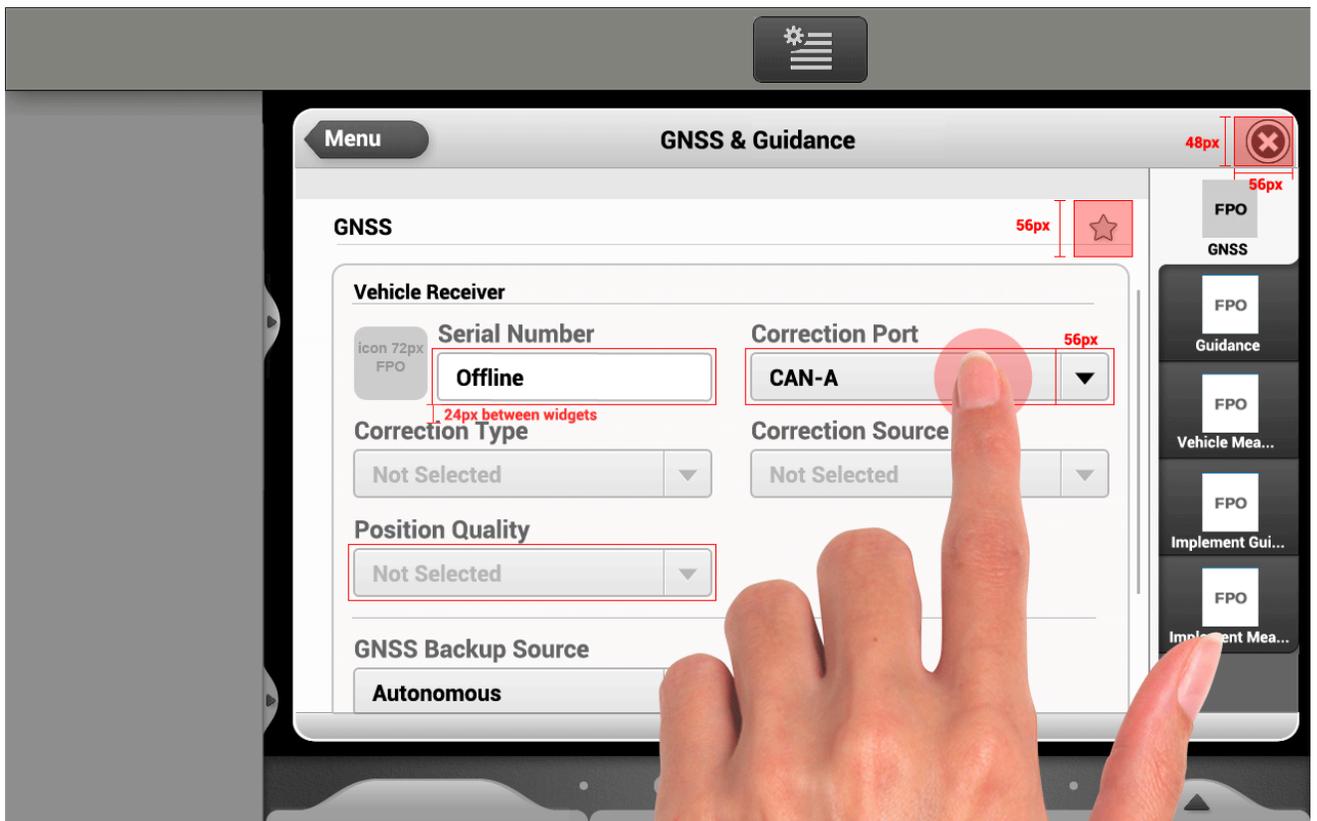
Figure 1. *Visual Example Touch Target Size.*
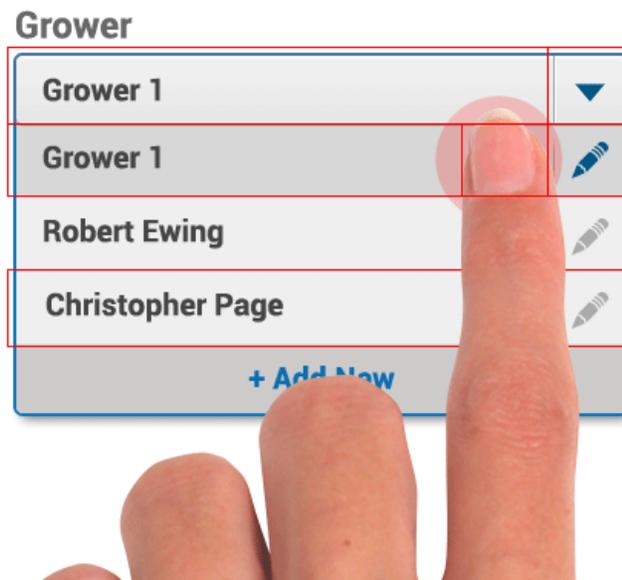
Figure 1.1. *Example Touch Target Size with Two Buttons Side by Side*

2. Command Buttons

Command buttons allow users to perform an immediate action.

Description

- Use buttons to initiate actions.

- On tap, the button shows a pressed state so the user knows the button press is registered.

- Action is taken upon release of the button

- If while tapping the button, the user slides their finger outside the touch area of the button, the action does not occur.

- Button can be of various widths to accommodate the text.

- Use single-word labels where possible.

Figure 2. *Example of Command Buttons in the Eagle application*

Label	Meaning
Yes	Used on dialog boxes when users have to give an Yes/No answer which does not mean closing or saving necessarily
No	Used on dialog boxes when users have to give an Yes/No answer which does not mean closing or saving necessarily
OK	When selected, confirms the choices and closes the dialog box or pop-up window
Save	Saves a group of settings, file or object
Done	Completes the task and accepts the changes
Cancel	Discards the changes
Close	Closes the popup window or dialog boxes by dismissing them
Back	Takes user to the previous page (i.e. in wizards and task flows)
Pause	Pauses an action or task without suspend
Next	Takes you to the next step in wizard or calibrations

3. Input Text Field

Allows users to type in or edit a text or numeric value.

- Text fields can be either a single line or a multi-line text box.

- A virtual keyboard appears on the screen when users tap a text field. Users can hide the keyboard by touching the screen outside of the text field. To bring the keyboard back, users have to touch on the input field again.

- Use auto-complete to help users enter data that is likely to be used repeatedly. Examples include user names, addresses, and file names.

- Display the Clear button (x) in the right end of a text field. Tapping it clears the content of the text field.

- When a text field becomes unavailable, the label that describes it should show as unavailable as well.
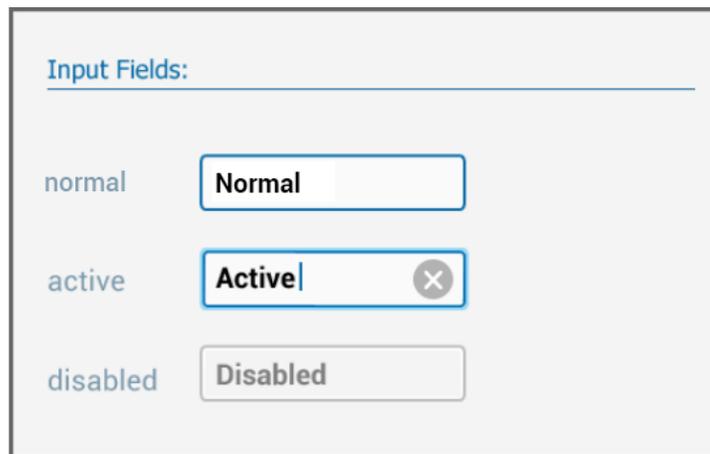
Figure 3. Example of a single line Input Text field
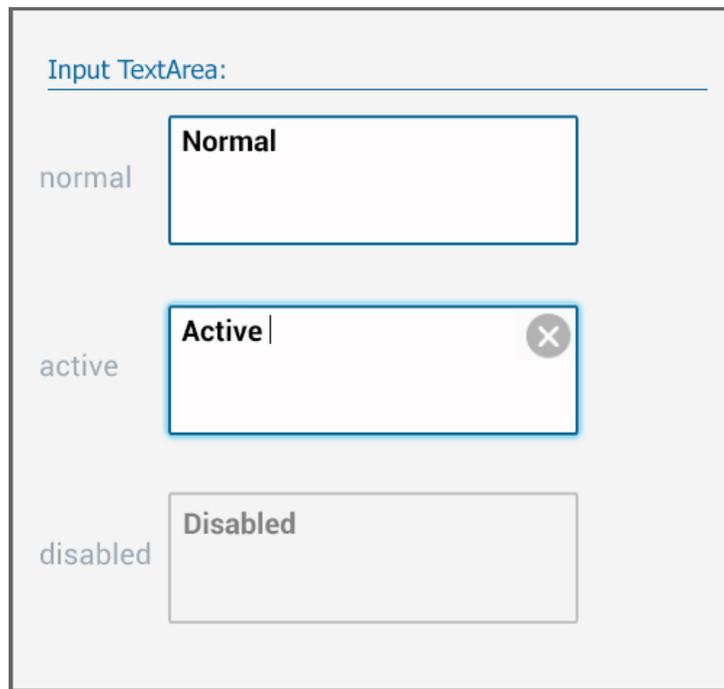
Figure 3.1. *Example of a multi-line Input Text Field*

4. Read Only Text:

Read only texts are non-editable text fields that simply display text.
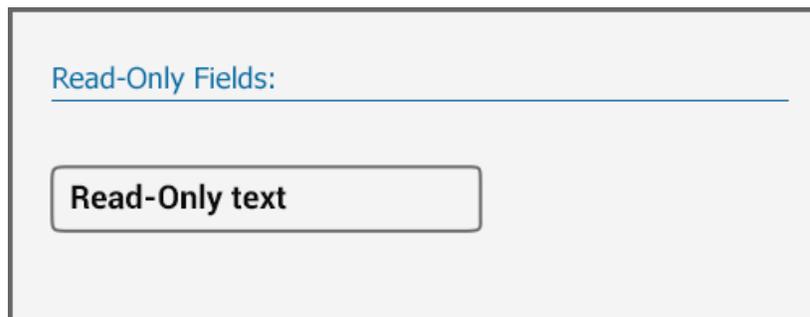
Figure 4. *Example of Read-Only field*

5. Labels

- All controls (e.g. text boxes, input fields, pick lists) need labels. Write the label as short as possible.

- Position the label either above, to the left or on the control

- Left align the label with the left edge of the control.

Figure 5. *Example of a Label on top of a Pick List*

6. Tabs

Tabs are used to alternate between views and allow the user to locate categorized content and features and give a clear indication of their current location inside the parent window.

- Tabs provide a selection between top-level sections and subsections of a feature.

- Each tab shows the name and the icon of the represented section.

- The active tab has a high contrast visual distinction (i.e. light gray color) between the inactive tabs (i.e. blue color).

- The tabs can be organized only in one row.

- The labels use Title-Style Capitalization: each word's first letter is uppercased.

Two types of tabs are used throughout Eagle Display: horizontal (i.e. GNSS status, Menu) and vertical tabs (i.e. runscreen and card detail pages tabs).

Vertical tabs on the detail pages can be scrollable if there is a need to fit more tabs. The label on the vertical tabs on the detail pages should be a short version of the Section name and should be specified in the technical requirements. If more than five tabs are used, a visual scroll indicator will appear to inform user of the additional tabs.

The runscreen vertical tabs do not have a label and should never scroll.

Figure 6. *Example of Horizontal Tabs - Menu*

Figure 6.1. *Example of Horizontal Tabs - GNSS Drop-Down Window*
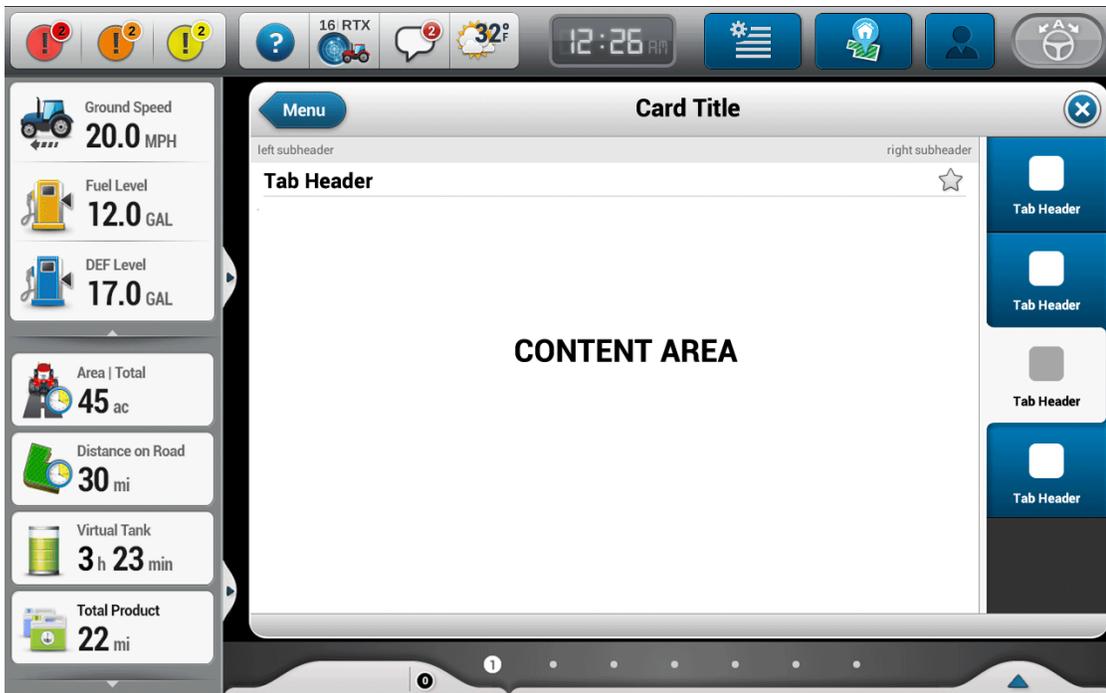
Figure 6.2. *Example of Vertical Tabs (Four Tabs) – Detail Page*
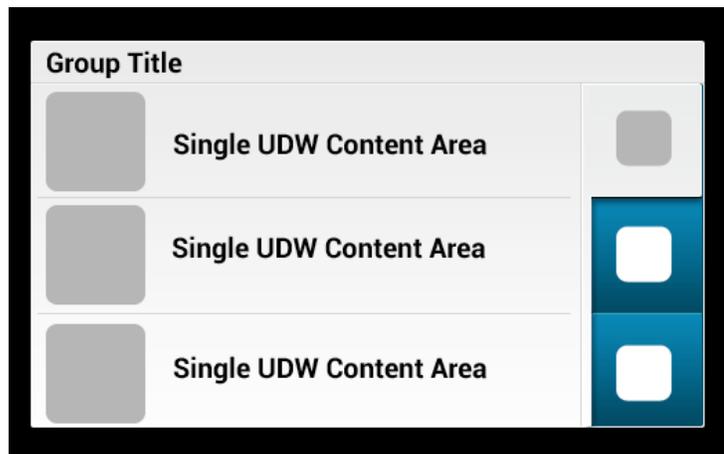
Figure 6.3. *Example of Vertical Tabs (Three Tabs) – Run Screen*

Best Practices:

- Try to organize the tabs in a logical order.

- Try to limit the number of tabs if possible, so that the users need not scroll.

7. Segmented Buttons

The segmented buttons control is a horizontal control consisting of multiple discrete buttons. The control provides a compact way to group several buttons together. It can be used to display closely related, but mutually exclusive choices one.

- The label must fit in a single line;

- Only one button is selected at a time;

- It can be used for selections of maximum 4 choices.

Figure 7. *Example of a Segmented Toggle Button*

8. Inline Errors

Inline errors are messages that appear in the context of the current UI surface instead of a separate window.

The related error help text is added below the field in question.
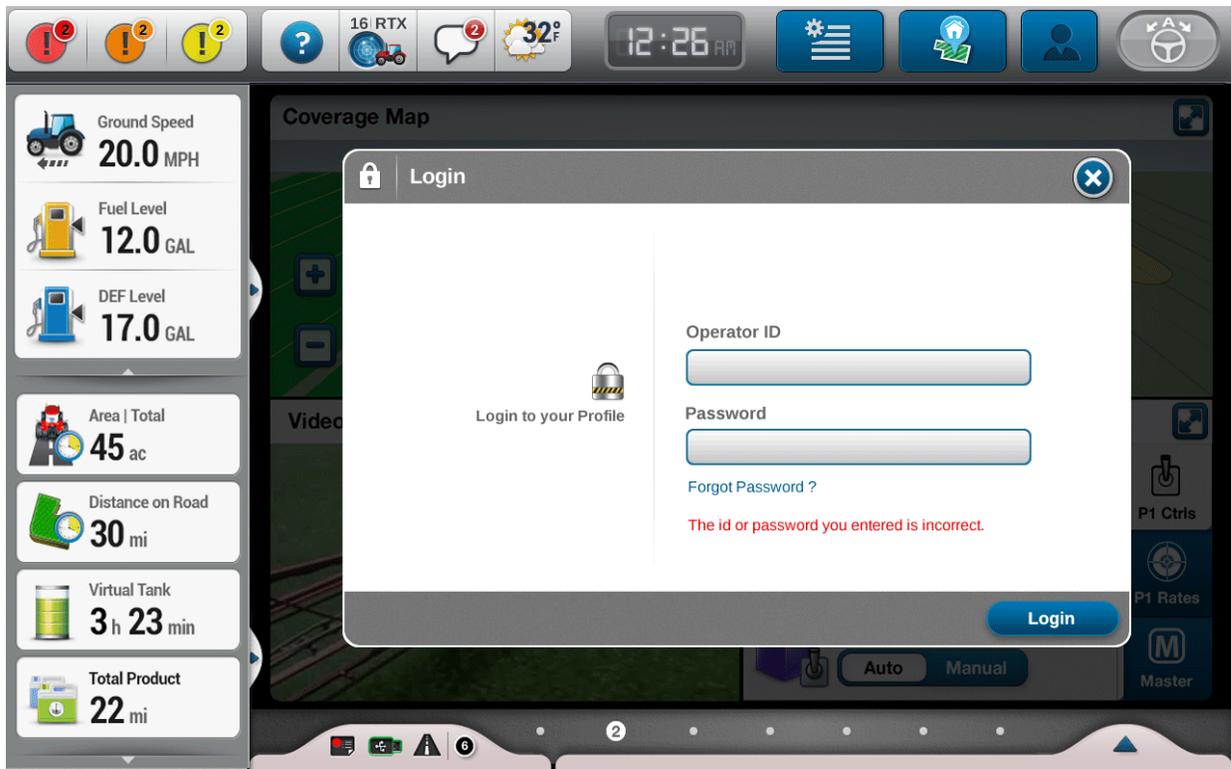
Figure 8. *Example Password Validation Error*

9. Pick List

A pick list is a control that allows users to make a single selection from a list of two ore more items. The main consideration to use a pick-list would be the case when the user has an idea of what item he needs to select from the list, what's left is to find and select it (i.e. selecting the language out of a long list, the user knows ahead what he is looking for and does not give too much consideration for the other choices). The selected item is always visible at the top. By clicking on the expand arrow icon the user can access the rest of the list items and make further selections.

Description:

- A pick list should always have a selected item, which shows in the pick list's closed state.

- For the use cases when the user did not make a selection yet, either the default selection, defined in the technical requirements, should be used or, when a default cannot be established, the 1st available item or an instructional selection (i.e. "- Select item type -") should be used.

- When the list items have long character strings, on opening, the pick list will expand horizontally to accommodate the items' full width.

- In the expanded mode the currently selected item should be highlighted. If the item is not positioned in the first available view within the list, the list view should jump in such a way as to center the currently selected item in the middle of the pick list.

- When the users tap to make a new selection, the newly selected item will show in the header of the pick list and the pick list will be closed.

- The way of sorting the list items should be defined in the technical requirements (i.e. alphabetically, by date or it can have a predefined order).

- The pick lists should be used when selected option is far more important than seeing all the alternatives.

- There are two different types of Pick Lists used in the Eagle environment.

Standard Pick List

- A standard pick list contains only a list of pre-identified items either created during setup or provided by the system. A user can make a single selection from within the expanded pick list.
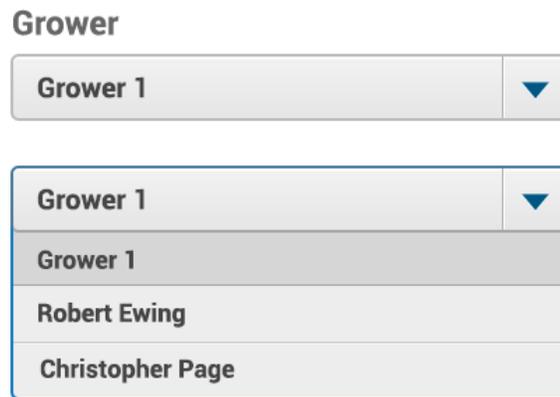
Figure 9. *Example of Pick list*

Combo Pick List

- Based on technical requirements, a combo pick list may or may not contain a pre-defined list of items, but it will include additional functionality/actions to allow the user to manage the list of item (i.e. Add, Edit, and Manage. Future development may require additional functionality inside the pick list, which are to be added to this section).

- When user taps the Edit or Add New button, the keyboard appears on screen and user can edit the selected item.

- The Manage button brings a pop-up window where users can make additional changes.
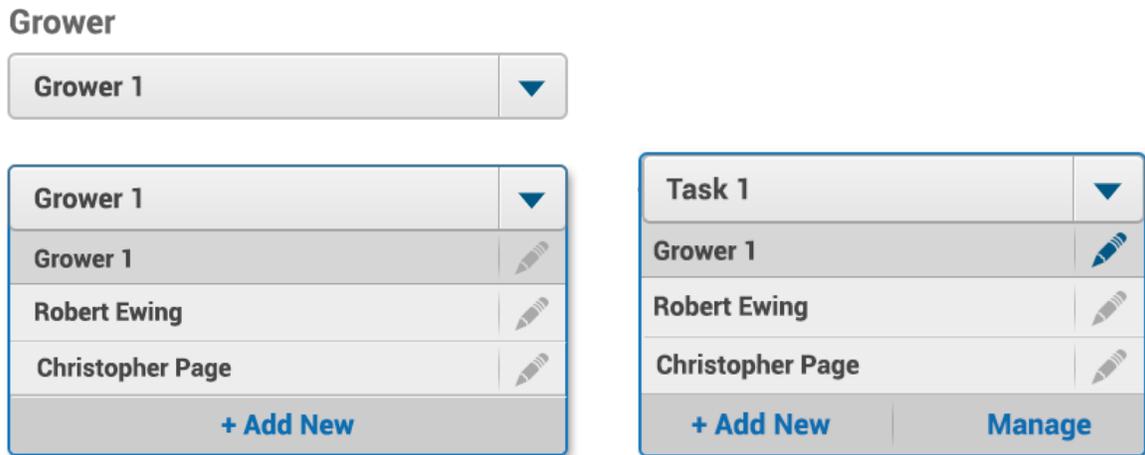
Figure 9.1. *Example of Combo Pick List*

10. Radio buttons

Radio buttons should be used when users have to make a single selection out of a list of two or more items while the whole list of options must be visible. A major clue to use this widget would be the use case when the user has to analyze all the options before making a choice (i.e. choosing the unit system).

- One item of the list is always selected; None can also be one of the list items

- The Radio buttons are placed to the left of the list items.

- Radio buttons shall not be used within run screens on the Eagle display due to space constraints.

Best Practices:

- Consider using a pick list if you do not want to draw attention to the options, or you do not want to encourage users to make changes. A pick list focuses on the current selection, whereas radio buttons emphasize all options equally.

- Do not use radio buttons to display long list of options. It becomes hard for users to scan the options for more options, use a pick list.
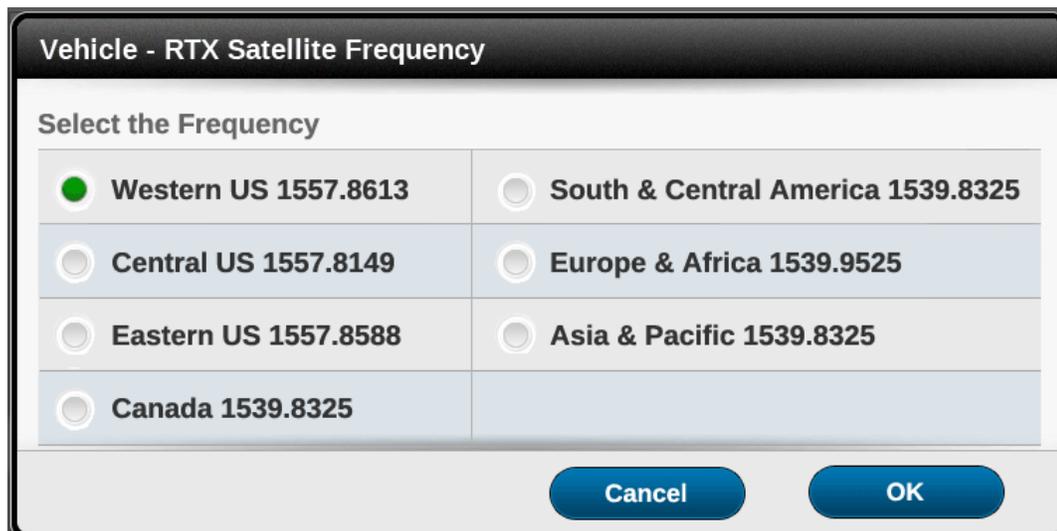
Figure 10. *Example of Radio Buttons in a List*

11. Check Boxes

Check boxes allow users to select multiple items within a list of multiple selections.

Description

- As a rule check boxes have two states, checked and unchecked.

- There are use cases when each item of a list of Check Boxes contain another list of check boxes. The parent check box item will appear as checked/unchecked only if all the check boxes inside are checked/unchecked, though when this is not the case, there is a need to define a third state, "Partially checked", for the parent checkbox to show that only some of the items inside are selected.

- The check boxes are placed to the right of list items

- Unlike single selection controls such as radio buttons and pick lists, users can select any combination of options in a group of check boxes, including zero, one, or several.

- The check box should not be used as an on/off control; use an on/off toggle switch instead.
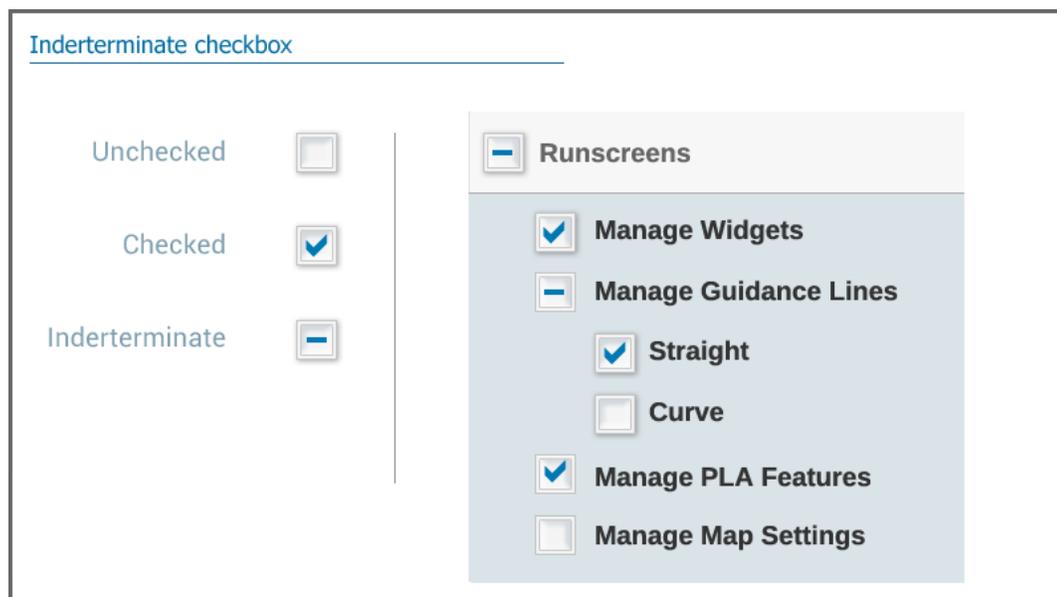
Figure 11. *Example of Partially Checked Checkboxes*

12. On/Off Toggle Switch:

Use a toggle switch for binary operations that become effective immediately after the user changes it.

- Use a toggle switch for an option that can be turned on and off.

- Users can tap to alternate the options or drag the slider to the left or right to change the state of the switch.

- *Examples:*

- On/Off: Used to turn services or hardware components on or off

- Lock/Unlock: Used to enable or disable a section
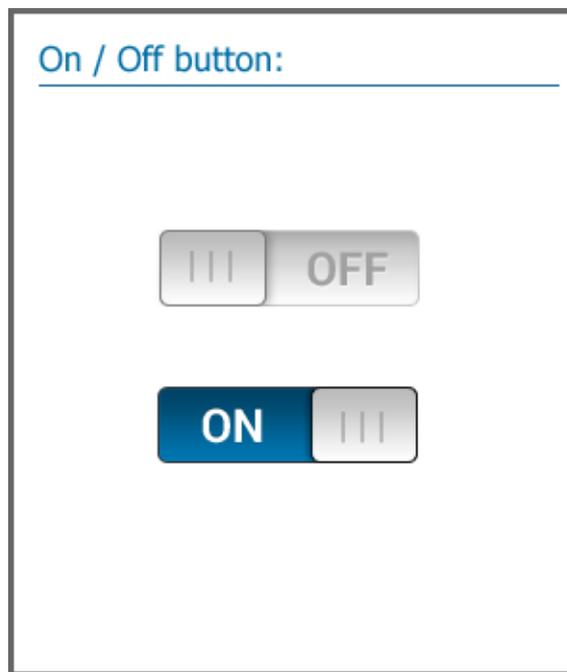
Figure 12. *ON/OFF Switch*

13. Progressive disclosure controls

Allows users to show or hide additional information including data, options, or commands.

The goals of progressive disclosure are to:

- Simplify a UI by focusing on the essential, yet revealing additional detail as needed.

- Simplify a UI's appearance by reducing the perception of clutter so that user can easily access the information.

Figure 13. *Examples of Progressive Disclosure Control used in Eagle.*
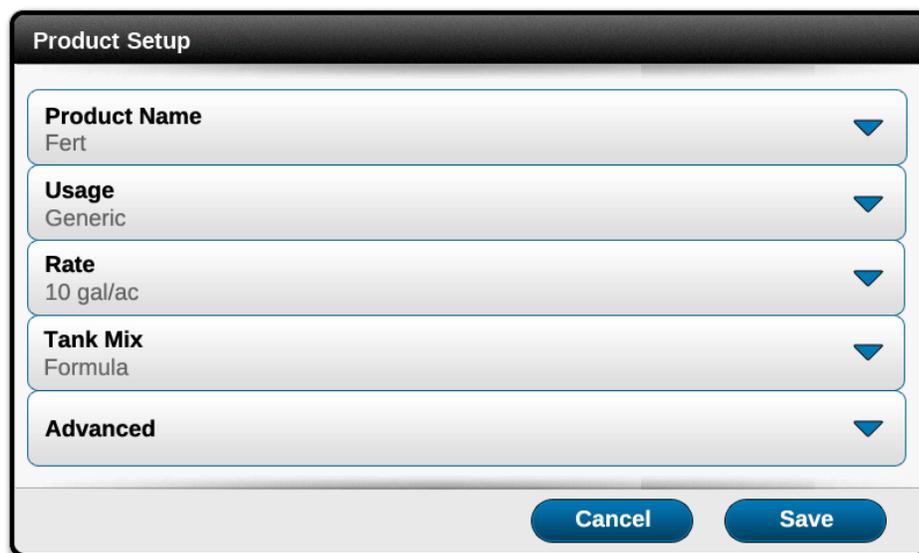
Figure 13.1. *Example of Progressive Disclosure in a Selection List*

Example: Permissions display the important categories all the time, but uses Progressive Disclosure to display details on demand of each category.
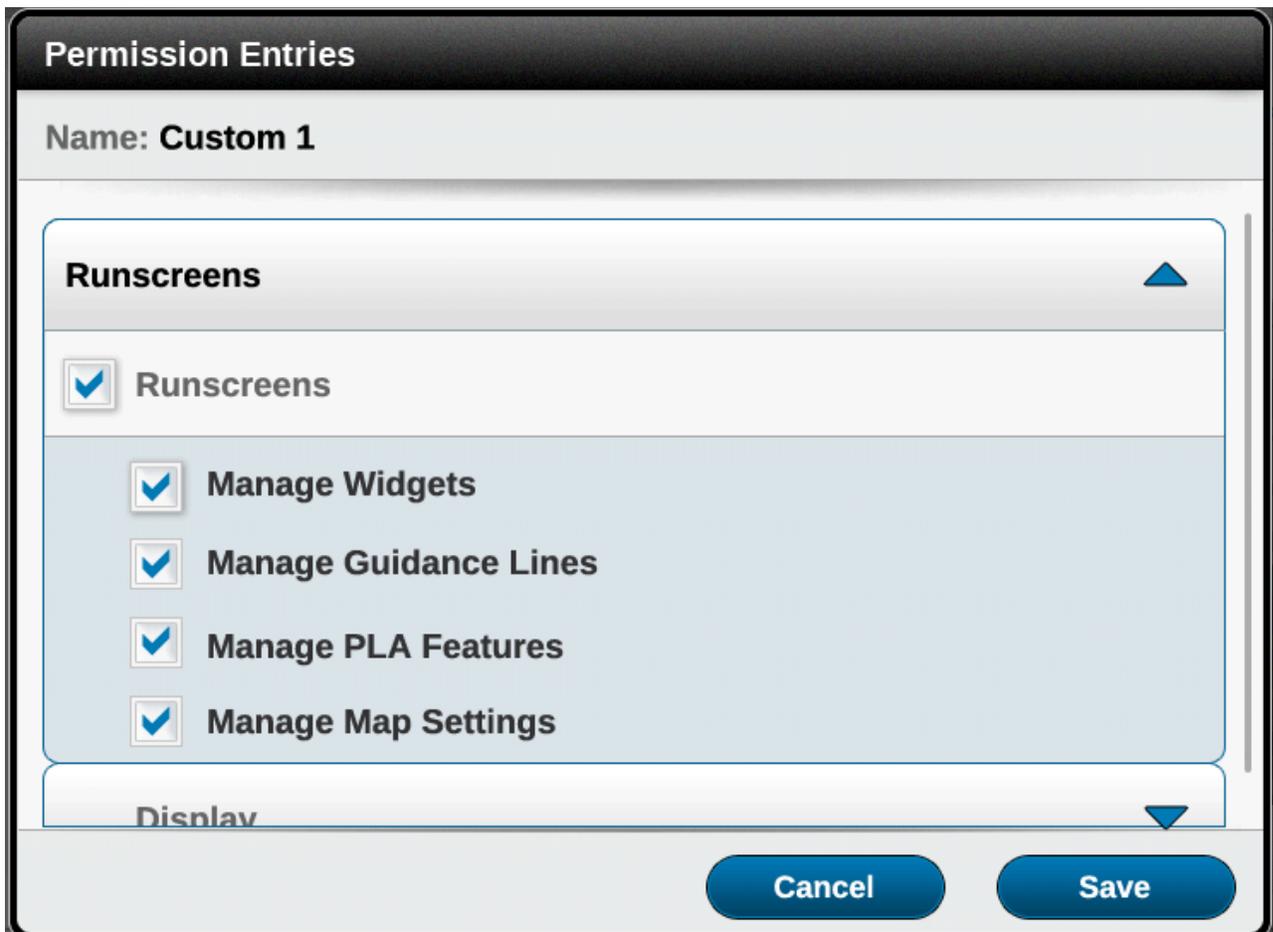
Figure 13.2. *Example of a Main Category and a Selection List:*

Best Practices:

- Use progressive disclosures when users need to see the information in some but not all scenarios.

- Do not use progressive disclosure if users are not likely to hide it.

14. Stepper

Stepper is a control, which represents a value with a plus and minus button on the left and right side of the control. Some steppers can be accompanied by an input field that allows users to customize increment value. The newly entered increment will be effective immediately.

- By tapping on the plus and minus buttons users can adjust the value in increments which are defined by technical requirements,

- User can adjust the value exponentially by long press on the buttons

- The stepper control shall expand horizontally, if required, in order to accommodate the value and the unit, though a minimum and a maximum width should be defined.

Eagle has five types of Steppers:

1. Stepper where the value can be tweaked by tapping on plus and minus as well as entered from the keyboard.

Figure 14. *Example of Stepper One.*

2. Stepper where value can be adjusted only in predefined increments by tapping on the plus and minus

Figure 14.1. *Example of Stepper Two.*

3. Stepper where the value can be entered from the keyboard as well as tweaked in user-defined increments. A default increment should be established in the technical requirements.

Title

| − | 1.5 in | + | 🔲 0.5 |

Figure 14.2. *Example of Stepper Three.*

4. Stepper where the value can be tweaked in user-defined increments. A default increment should be established in the technical requirements.

Title

| − | 1.5 in | + | 🔲 0.5 |

Figure 14.3. *Example of Stepper Four.*

5. Stepper in an unavailable state as a result of previous setups or selections

Title

| − | 1.5 in | + |

Figure 14.4. *Example of Stepper Five.*

15. Sliders

The slider is a three-segment control, which makes it possible to select a value from a continuous or discrete range of values by moving the slider thumb or by pressing the right and left arrows. The smallest value is to the left, the largest to the right.

Eagle has multiple types of sliders

- Interactive Sliders

    1. General Slider
    2. Reference Point Slider

- Read Only Sliders

    1. Color Spectrum Slider
    2. Read Only Slider

15.1. Interactive Sliders

General Slider:

General slider is a three-segment control. It contains the current value, left and right range of values and interactive arrows to increase or decrease the value.

- Users can press the right and left arrows, which increases the values by predefined increments specified by functional requirements.

- Users can adjust the value exponentially by long press on the buttons

- Users can drag the handle to increase or decrease the value. While dragging, a hover bubble will follow along with the user's finger to display the input value

- Users can tap on the slider and the value will be adjusted correspondingly.

- Current value is displayed in the middle of the slider

Figure 15. Example of General Slider

15.2. Reference Point Slider:

Reference point slider is a three-segment control slider with a visual indicator of programmed settings or system default that the user would need to see and revert it to the previous value.

- Reference slider will reflect the default value with a "red" color

- Reference slider can display a default value in "red" and a desired value in "green" color

- In some use cases reference sliders can also include a "Save" or "Revert" button, where the visual indicator will change to reflect the newly saved value.
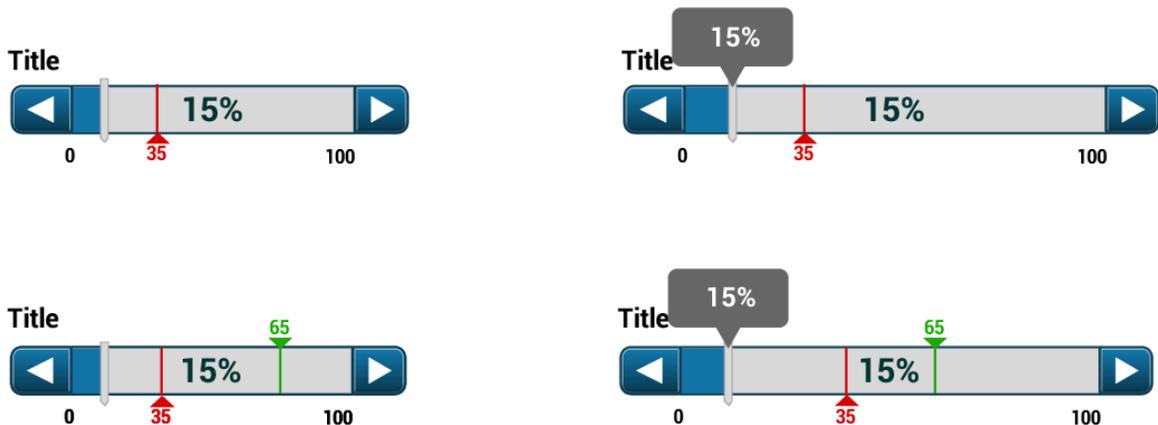
Figure 15.1. Example of Reference Point Sliders
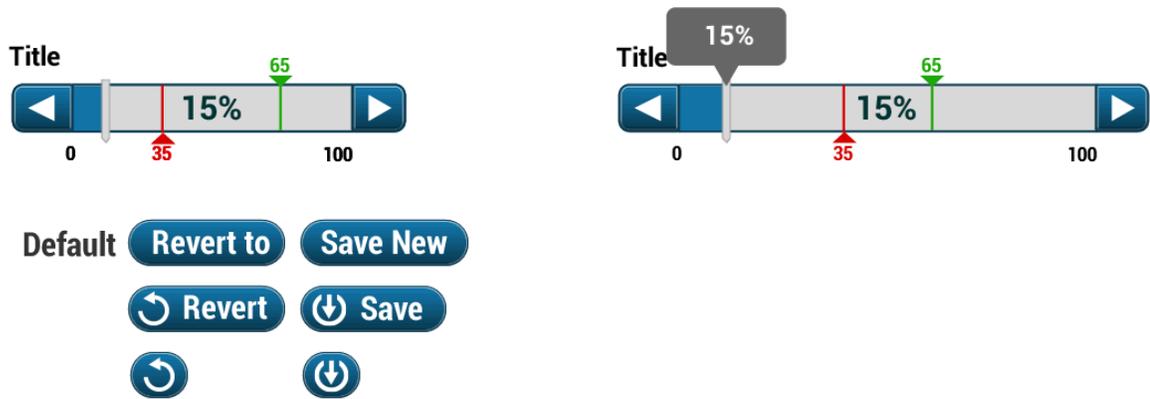
Figure 15.2. Example of Reference Point Sliders

15.3. Read Only Slider/Gauge:

Color Spectrum Slider:

The Color Spectrum Slider is used when we have to deal with a value within a range and quality values such as good, medium, bad serving as secondary ranges. Displaying the quality value is as important or even more than the value itself. By default green is used for good and red for bad and the entire spectrum in between.

The pointer shows in which secondary ranges the value is located.

Figure 15.3. Example of Color Spectrum Slider

15.4. Read-Only Slider:

The Read only slider displays the current value along with the minimum and maximum values. The minimum value is to the left; the maximum value is to the right.

Converge Level

0.501 m

0   1000

Figure 15.4. *Example of Read-Only Slider*

16. Progress Bar

A Progress Bar is a read-only widget used to visualize the progression of an extended operation, such as a download, file transfer, or installation. Sometimes, the graphic is accompanied by a textual representation of the progress in a percentage format.

A progress bar can display additional information such as a summary of completed steps/events.

- The label, is such exists, should be displayed above the widget.

- Make the status text short; just inform the user about the process by one or two words (i.e. "loading" "Copying")

- If the percentage of progress is not known, indicate a continuous action, ellipsis after the label (i.e. Loading…).

Loading (72.5%)

0   100
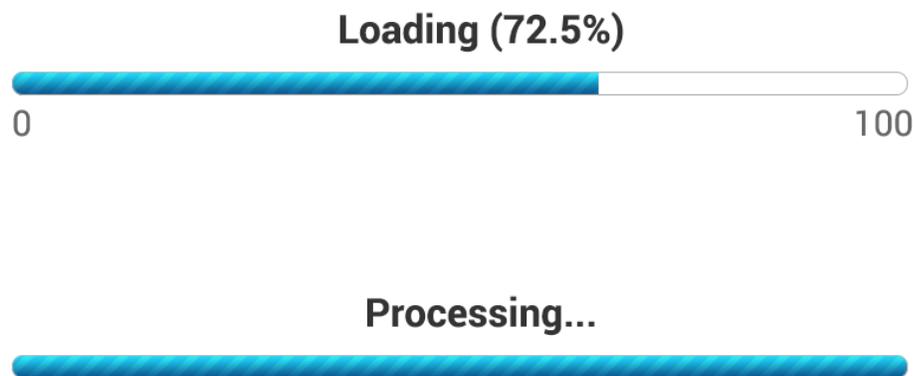
Processing…

Figure 16. *Examples of Progress Bars*
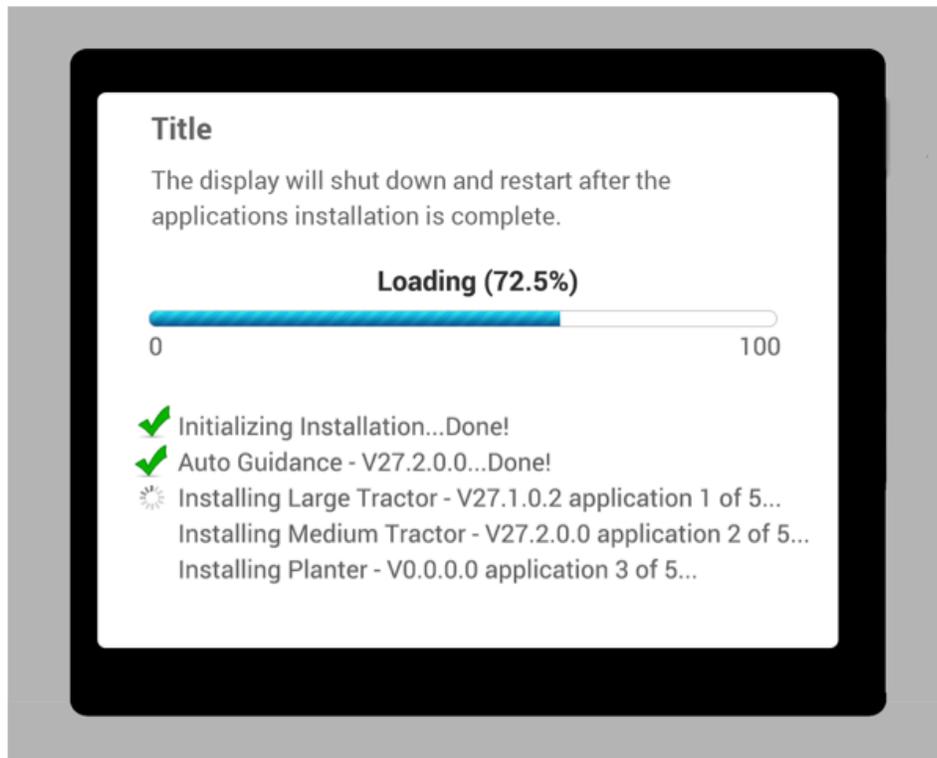
Figure 16.1. *Example of a Progress Dialog*

17. Process Indicators:

Process indicators are read-only visual cues that indicate that a process is underway. It provides a step-by-step feedback to the user, as the process could be lengthy.

Process indicators represent data differently than a progress bar, it shows the step-by-step operation, such as calibrations, wizards.

For calibration use case, a segmented bar that "fills" the dots from left to right indicates the progress.
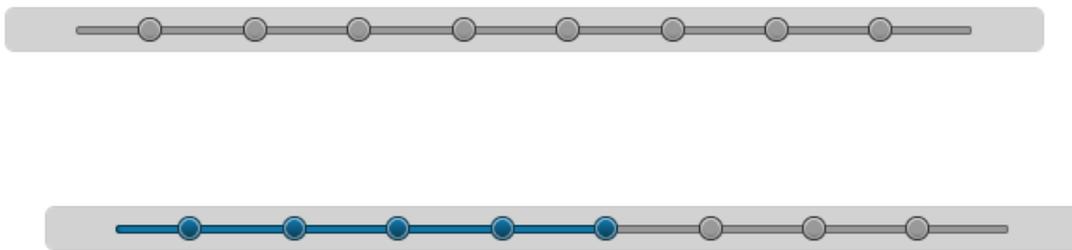
Figure 17. *Example of Process Indicators*

- Process indicators shall not be used for one or two steps processes.

- Process indicators are a sequence for a predetermined number of steps. The system shall configure the setup upfront so the wizard already knows the number of steps.

Exception: In Eagle non-predefined steps will also be used to show process.

18. Activity Spinner

The activity spinner is used in the situations where the extent of the task is unknown or the progress of the task cannot be determined in a way that could be expressed as a percentage. This bar uses circular motion to show that progress is taking place.

Activity Text...

Figure 18. *Example of an Activity Spinner*

- The progress bar should be used for actions that may take longer than ten seconds.

- Activity spinner should be used if the process might take less than ten seconds.

19. Pickers

Date Picker

A date picker displays components of date and time, such as hours, minutes, days, and years. Tap each menu to choose a value.

Figure 19. *Example of a Date Picker*

Time Picker

This displays the hours, minutes, and AM or PM. Tap to choose a value.

20. Language, Localization

Localization refers to the process where a product is adapted to be used in different countries without redesign. On the other hand, localization addresses how a product may be tailored for a specific country, region or culture by making it linguistically and culturally appropriate (i.e. different unit system or reading from left to right)

Design for Text Expansion and Contraction

Text may scale or contract in different languages. To account for that, the general rule is to plan for text expansion of up to 30% to 40% when going from English to German, which is one of the more extreme examples of text expansion. Going from English to Spanish or French will probably only result in a 10% to 20% expansion. To safely design for any target language, a goal of 35% expansion is a good guideline.

For example, "Search" in English translates to "Rechercher" in French, which is nearly twice as long. Allow space for an additional 40% or more of room for expansion in the menu.
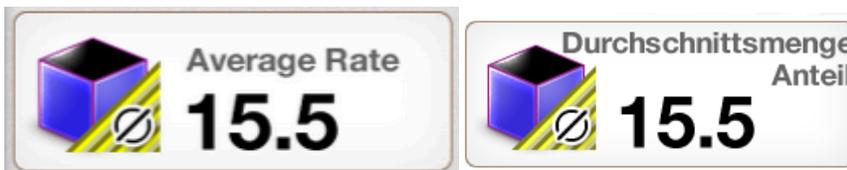
Figure 20. *Examples*

Design for Ellipsis

Eagle display will use ellipsis when the localization may exceed the available space. Ellipsis shall be used as follows:

- In use cases where we have one long word the text shall be truncated at the

- (e.g. Product Management shall be Product Manage…)

- In use cases where we have two or more words the text shall be truncated in the middle

- (e.g. John Stratton's Farm shall be John…farm…)

ICONS

Icons are used throughout the Eagle user interface to represent equipment, vehicle functions, processes and tasks. It is important to design effective icons that communicate their purpose.

Use a common design style within icon types. Repeat common characteristics but avoid repeating unrelated elements.

Follow ISO Standards closely or use real-world (rather than metaphorical) imagery when no ISO icons exist. Literal representations communicate concepts more effectively than abstract symbols.

Icons should be designed as a set when constrained by a function or system (i.e. hitch or guidance recording icons); consider their relationship to each other and to the user's tasks.

Screens and windows not identified under the icon section of this document shall not have icons.

When a new icon is created, all instances where the icon would be used in the system should be specified and designed for.

1. UDW Icons

(This is the first time UDW makes an appearance. It needs to be defined in the Dictionary of Terms.)

UDW icons will be used in Run Screen, Left Hand Area (LHA) and LHA drawers. Only one size of the icon shall be created. On LHA the icon should be used at 84% (automatic resizing). Vehicle status related icons could be dynamic (ex. fuel level). Some icons can have additional info overlay which needs to be created as a separate graphic asset (ex: X when not available, a droplet to show the type of product). Each type of overlay will have requirements concerning position (ex: above, top right or bottom right corner etc.) and animation (ex: the flashing X on top of the icon). See example below:

Figure 1. *Example of UDW Icon*

Color: For UDW icons, use colors drawn from the industry standards and brands, based on ISO and common practices. The colors should complement and emphasize the value it represents. If the icons use a representation of a vehicle or equipment the color shall match the brand and the icons must be created for all possible brand related color schemes.

Figure 1.2. *Example of UDW Icon Color*

Run Screen Icons: 72x72px
LHA: 60x60px (resized automatically at 84%)

Figure 1.3. *Example of Run Screen Icon*

2. Status Icons:

Status icons will be used in the bottom status panel. Two icon sizes should be created, one for inside the panel and a smaller version of it for the handle.

Color: gray scale palette shall be used as the base color scheme with an accent color to highlight the action.

Size: 72x72; 60x60px; 28x20x

Figure 2. *Example of Run Screen Icon*

3. Tabs Icons

Used in Run screen and Card Manager.

These icons have a monochromatic and flat style. Two states must be designed: selected and regular. These icons shall also be used for Shortcuts in the bottom right panel.

Color: The regular state should be white; the selected state should be gray scale with the starting basic gray #222222.

Size for Vertical Tabs: 40x40px

Figure 3. Example of Tabs Icon

Size for Horizontal Tabs: 64x64px

Figure 3.1. Example of Tabs Icon at 64x64px

Size for Shortcuts: 64x64px

4. Card Icons

Used inside Menu.

Color: For Card icons use colors drawn from the industry standards, based on ISO and common practices. The colors should complement and emphasize the group it represents.

Size: 72x72px

Figure 4. *Example of Shortcuts Icon*

5. Expand/Collapse Section Icons

Shall be used in the section header to further inform the user of the content within.

Color: use colors drawn from the industry standards, based on ISO and common practices. The colors should complement and emphasize the section it represents.

Size: 72x72px

System Icons

These are icons that allow users to identify or engage in different system actions such as edit, add to shortcuts, save/delete, lock/unlock, add/remove from shortcuts, etc. These icons can have different states such as Inactive, Active, Pressed, Regular, or no interactive state.

Color: use colors drawn from the industry standards, based on ISO and common practices. The colors should complement and emphasize the section it represents.

Size: icon size can vary depending on the layout on which it belongs to. Example Undo on Shortcuts drawer and on the Edit run screen mode can be of different size. The minimum size of the system icons is 48x48px.

Figure 5. *Example of System Icons*

6. Alerts/Faults Icons

Priority: There are three priority fault icons (high, medium, and low), which are color coded (Red, Orange, and Yellow) accordingly. These priority icons appear in the top bar, drop down, history and fault alert windows.

Figure 6. Example of Alerts/Faults Icon

Fault Related: Equipment, Task, or Functions that require warning or failure notifications. These icons can be found in the top bar expandable content and fault alert windows.

Figure 6.1. Example of Fault Related Icon

Color: Use colors drawn from the industry standards, based on ISO and common practices. The colors should complement and emphasize the value it represents.

Size: 72x72px

7. Top Bar Buttons Icons

Consists of icons contained within the fault and actions panels as well as administration buttons. Typically these icons exist within a button or panel that has multiple states of interaction which can be reflected in the icon as well.

Color: Use colors drawn from the industry standards, based on ISO and common practices. The colors should complement and emphasize the value it represents.

Size: 60x60px

Figure 7. *Example of Top Bar Buttons Icons*

CNH CONTACT INFORMATION

Please feel free to contact the CNH team with any questions about this document. Thank you!

Key Team Members

Elisabeth Thebaud
User Experience Designer
Elisabeth.thebaud@cnh.com

Bushra Ali
Visual UI Designer
Bushra.ali@cnh.com