Archived:Application main view usability

Archived: This article is archived because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The images and examples are very specific to a very old version of Symbian. This should be updated to be UI agnostic or to the current Nokia Belle UI. As this is unlikely, the article has been archived.

Introduction

The main view is arguably the most important screen/view of any mobile application. The user always sees the main view of the application first and only from there s/he is able to navigate to any other view or able to use any of the functionality provided by the application.

Design Considerations

While designing the application user interface, the UI designer and usability expert should brainstorm the points in the following sections.

View Hierarchy/Layout

Consider what sort of a view layout and hierarchy (shallow/deep) would best suit the requirements of the application. Depending upon the functionality provided, how they can be logically clubbed together?

The designer should prepare wireframes which would give them an idea of the intended application/UI flow. The wire frame would give them a better idea if the application should have multiple views or can the functionalities be achieved using a single main view only.

Another thing to dwell on is to minimize the amount of information/controls available on the main view of the application. The main view should be as uncluttered as possible to give it a more organized/neat look.

Navigation Pattern

Once the decision has been made on whether the application has to single/multi viewed, they can dig further on how the secondary views should be accessible from the main view. Whether a menu option would be more apt, or a multi tabbed application view gives the user better access to the functionalities. The navigation should be easy to understand and follow for the user. The user should always get an idea of where they are in the application and how to get to the other views of the application.

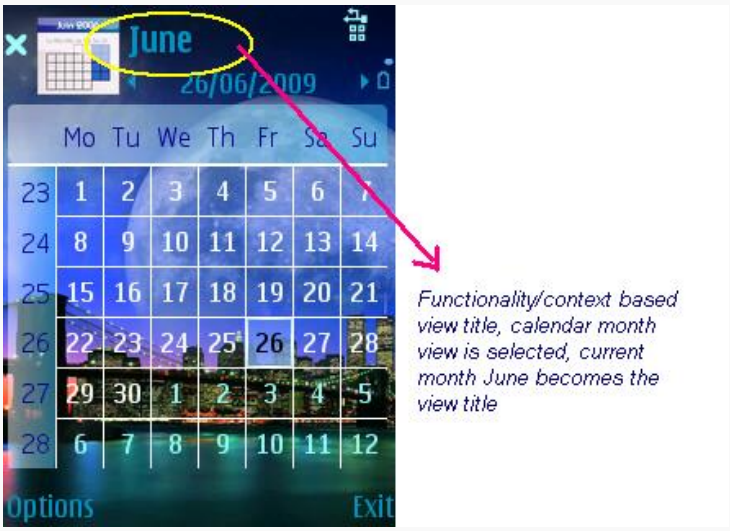
Homescreen usability guidelines

Since the main view is the soul of the application, focus should be put more on this view to get the navigation patterns/layout/look & feel/option menu etc right so that the user gets a good feeling when they interact with the application for the first time through this view.

View Title

The view should have a meaningful title which should be displayed prominently in the Status Pane of the application. The title of the view should make sense with the overall functionalities available on the view, or in case the views don't have separate titles, application title should be visible.

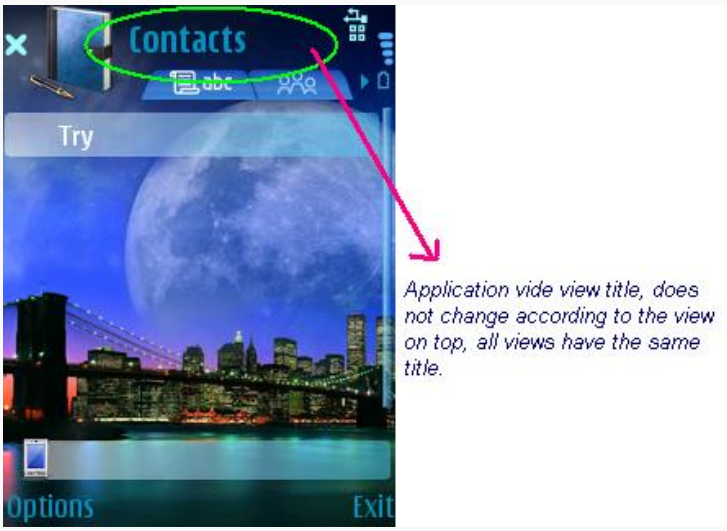
(C) Copyright Nokia 2013. All rights reserved.



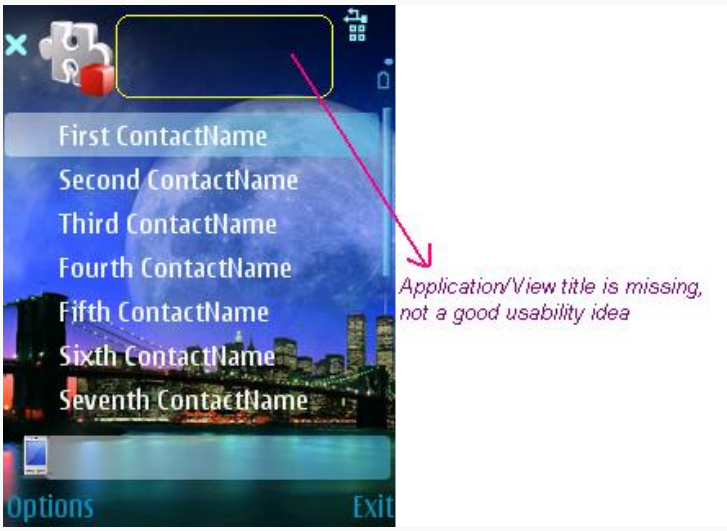
Context based view title

Functionality based title

(C) Copyright Nokia 2013. All rights reserved.



Application vide view title

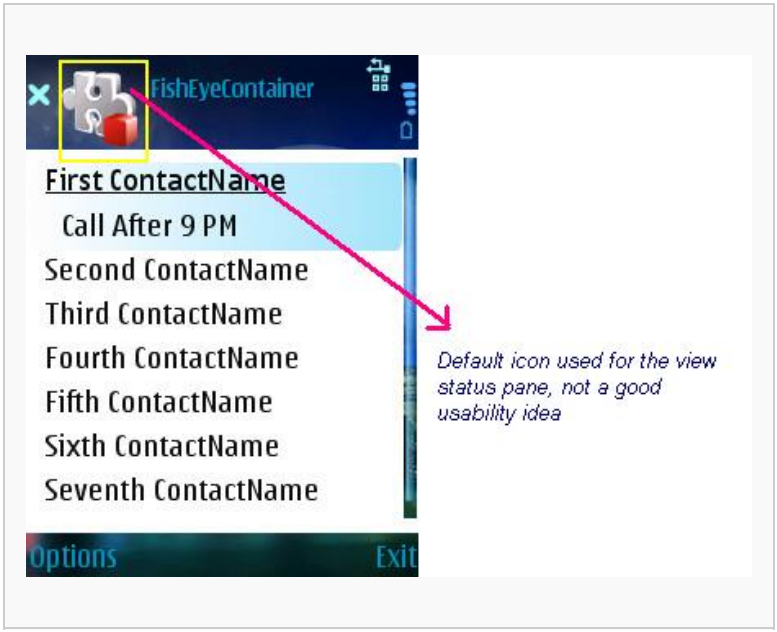


View title missing

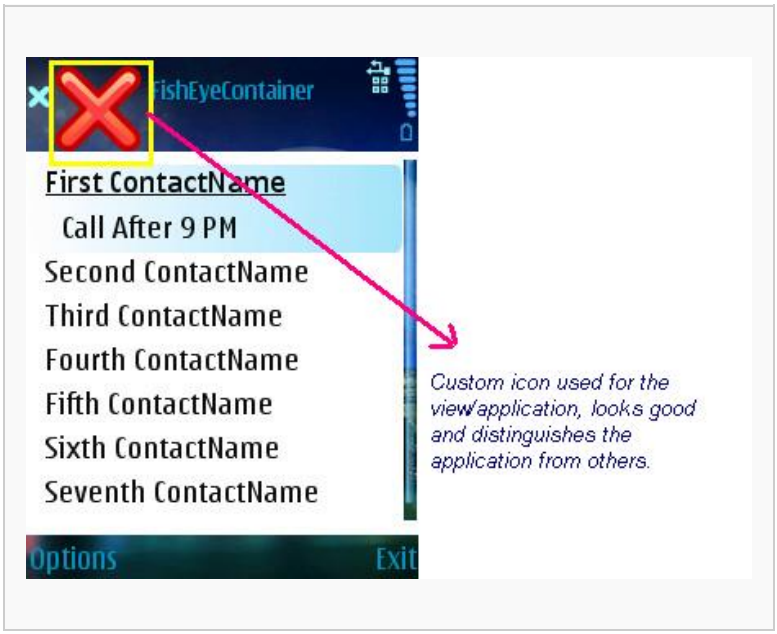
View/Application Icon

The application icon should be displayed on the status pane of the view.

(C) Copyright Nokia 2013. All rights reserved.



Default view icon



Custom view icon

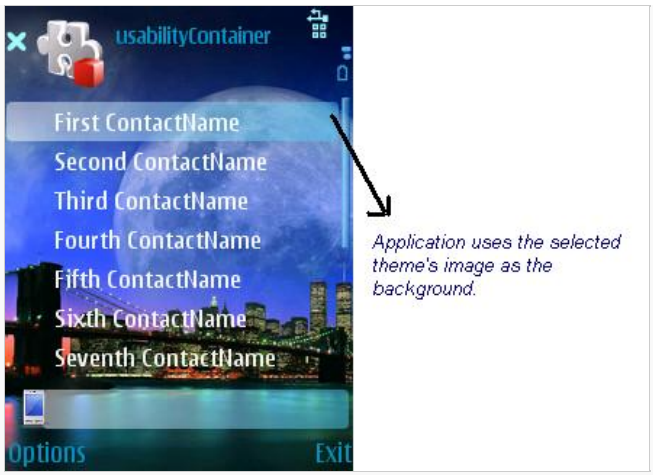
Meaningful graphics/color in background

The view background could either be:-

- The background image of the selected theme

Skins can be enabled for the application's main view, which would allow using the background image of the selected theme.

(C) Copyright Nokia 2013. All rights reserved.



Theme image as background

For skin implementation on S60/Symbian C++ check links:-

- Enabling Skin
- Enable Skin Support in Views/Containers
- Drawing skins in container
- Custom image deployed by the application

From a design stand point, the UI designer can also think of creating custom skins/images to be used as the background of the application view. In this case it is important to make sure that the color combinations are not too loud, and standout well against the default themes on the device.

Custom image as background

- Color pattern/animation implementation

The designer can also think about implementing the background using colors and animations. In this case it is important to make sure that the color combinations are harmonious and do not make it tough for the user to read the content being displayed.

(C) Copyright Nokia 2013. All rights reserved.

Wrong color combination for background

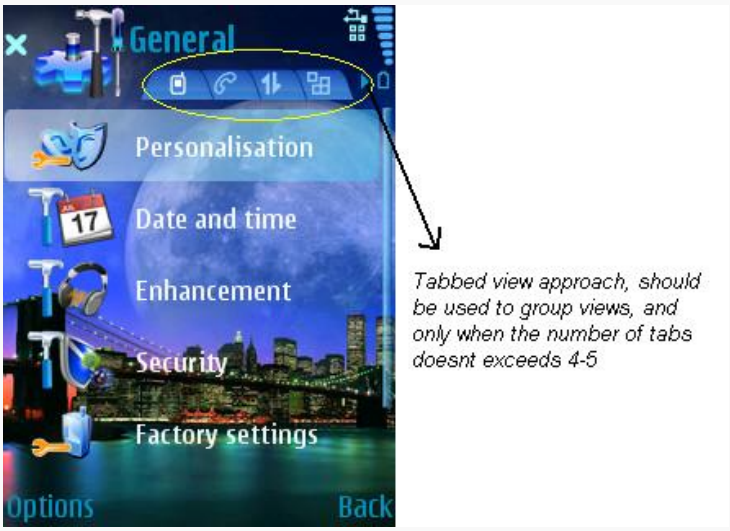
Right color combination for background

In case of custom drawing/redrawing of the view background, care should be placed to avoid any flickering as it doesn't give a good experience to the end user. To avoid flickering, double buffering can be used as a possible mechanism, details from Avoid flickering with double buffering on Symbian & Using double buffering in Java ME

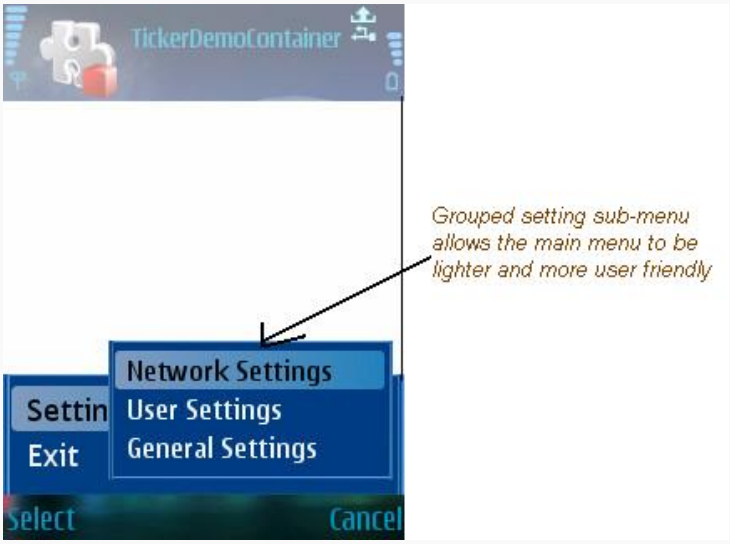
Consider tabs

One can think about providing icons/ tabs to access important views of application, in case the number of tabs does not exceed 4-5. If there are more possible views then that then accessing them through the option menu as grouped component could be a possible choice, or one can think about implementing grid based view as well.

(C) Copyright Nokia 2013. All rights reserved.



Tabbed View Layout



Access the views through the options menu

Options Menu

Application should show all the important functionality arranged category wise on homescreen's option menu. This way the user can easily come to know about the functionalities provided by the application and they can easily browse through the application.

Categories of functionality should be displayed in proper sequence; the main features of application should be shown on Top. Details can be had from Options menu Usability

No deep hierarchy

Should use a wide hierarchy instead of deep i.e the navigation pattern should be flat with multi views each accessible with one-two key presses, possibly from the main menu options menu. The views should not be hierarchical i.e. View1->View2->View3, where each view is accessible only through the hierarchy.

Be informative about the status of the application

The current status of the application should be clearly known to the user, if possible use some icons/text to denote the state of the application. For instance in case of a Voip application, it would make sense to denote the registration status to the user and so on.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.