WikiOnBoard Wikipedia Offline Reader - app showcase

Introduction

WikiOnBoard is a Qt Quick app which allows reading the Wikipedia without an Internet connection. This article provides an overview of the main elements of the app design.

The app can be downloaded from Nokia Store. The project including source code is hosted on github. Further information is available here.

Symbian Qt Quick Components Usage

WikiOnBoard has originally been developed using a Qt widget UI. The Widget UI version is still available in Nokia Store for mobile phones which don't support quick components.

To improve useability for newer devices the UI has been changed to Qt Quick Components.

A QML TabGroup allows easy switching between the three main functional areas:

- The Library page which allows opening wikipedia offline files and switching between them
- The Search page which allows searching for an article
- The Article page which allows reading the article

All views have a menu with different items. One common item is exit. The Article page has a back button to switch to the last article.

Library Page

The Library Page uses

- Flickable/ListView/ListItem/ListHeading/Custom Model

To display a list of Wikipedia Offline Files stored on the mobile phone

- Flickable/FolderListModel

To select a Wikipedia Offline File stored on the mobile phone

In addition, the "Download Wikipedia" button leads to a help page with information on how an offline version of wikipedia can be transferred to the mobile phone.

Index Page

The Index Page uses

- TextField

To enter article to search for. This is probably one of the biggest benefit of Qt Quick Components, because it enables split-view input easily.

- Widget based list embedded using QGraphicsProxyWidget/Kineticscroller

To display search results. This is due to reuse of existing component. Works well therefore for now not changed to Qt Quick Components

Article Page

The article page uses

- QTextBrowser Widget based embedded using QGraphicsProxyWidget/Kineticscroller

To display and navigate articles. Again reuse of existing component. Evaluation of porting to Qt Quick Components (QML WebView) ongoing, as described in more detail below.

Common Functionality

(C) Copyright Nokia 2013. All rights reserved.

Both the Download page and the article page use an adapted version of QueryDialog to ask the user for confirmation before opening an external link in the browser. The adaption has been done to support text wrapping, which is necessary because the link is displayed in the dialog.

The user is asked for confirmation for two reasons:

1. For an offline reader a major use case is that the user does not want to use an internet connection, for example because of roaming costs.
2. The app does not have control over the displayed content. For example, zim files exported from a wikipedia book, as described here don't show whether a link is external or internal. In these cases asking for confirmation improves useability significantly.

Design Guidelines

Problem Areas

Exit not via Back Button

The guideline to use the back button to exit the app would not work well with this app. The back button is used to go back to previously visited articles, just like in a web browser. Therefore using the back button to exit the app would on the one hand require a lot of clicks to exit, on the other hand it would be quite easy to exit the app by accident. The alternative of using it to exit only when the user is on search and index pages would be an inconsistent use of the back button, which is not desirable in terms of useability.

Furthermore, being able to exit the app from the menu is still common in recent apps, for example in the Nokia Belle Browser.

Therefore this approach has been chosen in WikiOnBoard.

Widgets Reuse

The approach to reuse existing widgets via QGraphicsProxyWidget/Kineticscroller works very well on Symbian, but unfortunately is barely feasible on Meego. The reasons are that the kinetic scrolling does not work and the performance in particular on the article page is pretty bad.

A potential workaround is to use QML WebView instead of QTextBrowser. Unfortunately this also imposes some challenges:

- Loading resources, in particular images, programmatically is more complex than with QTextBrowser. In the trial QNetworkManager is used for this purpose.
- Load times are longer. This is no surprise, because QTextBrowser is very fast, but only supports a limited HTML subset, while WebView is a full web browser.
- Some bugs need workarounds. It is particularly noteable that scrolling to anchors does not work (see bugreport), and that the transparent background is a bit too transparent (see bug and the workaround used in WikiOnBoard).

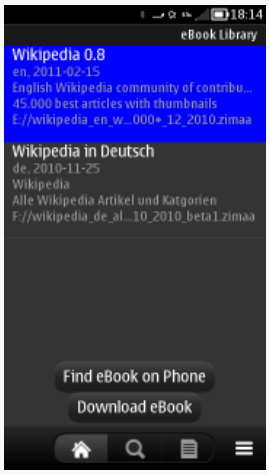
However, in summary switching to WebView in future seems to be promising. The source code for the trial is available here.

Start-up Time

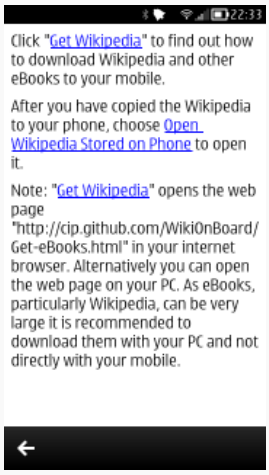
The start-up time is significantly longer for the Qt Quick Components version than for the widget version. QML Loader components have been used to reduce the start-up time.

Furthermore, a splash screen has been added to reduce the useability impact of the longer start-up time. This is in particular important on Belle, because the start-up animation is, in contrast to Anna, disabled.

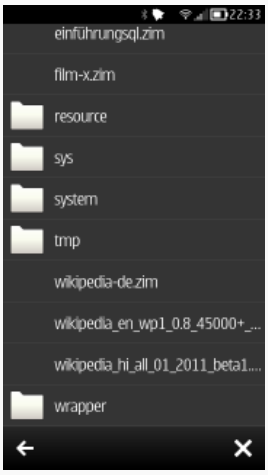
Screenshots



Library Page

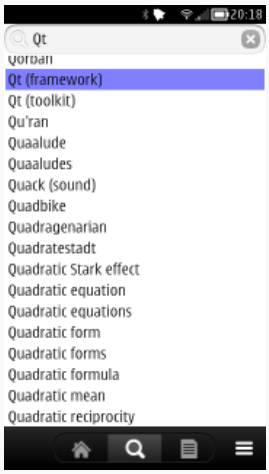


Download Help

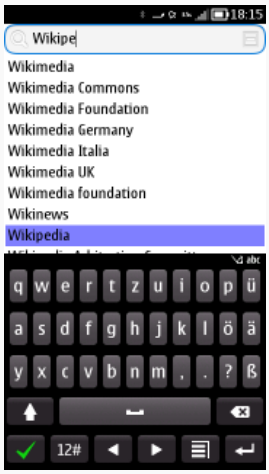


File Selection

(C) Copyright Nokia 2013. All rights reserved.



Search Page



Search Page with Keyboard

Article Page

Article Page Menu

Store Link

This app is available in Nokia Store.

Note: This is an entry in the Symbian Qt Quick Components Competition 2012Q1

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.