

XIDE User Manual

Document Name	XIDE User Manual
Editors	Evgenia Samochadina, samochadina@gmail.com
Version	Version 1.3
Date of Last Change	01.07.2010

Table of contents

1	Introduction	3
1.1	Purpose of the system	3
1.2	XIDE target users	3
1.3	Objectives of this document	3
1.4	Other resources.....	3
1.5	Document overview	4
1.6	Legend.....	4
2	Getting started tutorial.....	5
3	Advanced tutorials	16
3.1	View management	16
3.2	XIDE concepts	17
3.2.1	Application.....	17
3.2.2	Web Page.....	17
3.2.3	Component	17
3.2.4	Container	18
3.2.5	Template Language.....	18
3.3	XFormsDB application	19
3.3.1	What is XFormsDB?	19
3.3.2	XFormsDB application structure	19
3.3.3	XFormsDB application in XIDE	20
3.3.4	Application Folder structure	20
3.3.5	File access rights and sharing.....	21
3.4	Published application management.....	21

1 Introduction

1.1 Purpose of the system

XIDE is a web-based visual editor for developing XFormsDB-based web applications. It is designed to help users to create and manage web applications, which solve their personal tasks. It provides different features to facilitate development process at all stages, e.g. wizard-based application creation, visual page content creation, management of published applications.

1.2 XIDE target users

XIDE is focused on *non-professional users*. It allows to create highly interactive application without writing single line of code or having technical knowledge of XForms or XFormsDB at all. In order to do this, XIDE implements the idea of reusable components. There is a database of predefined components; each of them provides some small functionality and can be easily customized and built in to the web application.

However, more *professional users* are also taking into account. XIDE provides functionality to improve the process of application development in comparison with doing the development in plain text editor or IDE.

1.3 Objectives of this document

This user manual is designed for users of XIDE. It explains XIDE basics and allows novice users to get familiar with XIDE. It covers basic workflow and discusses more advanced features.

Information from this manual is also represented in the video format. The videos can be found on the XIDE welcome page <http://testbed.tml.hut.fi/xide/> and demonstrate how to start with XIDE and how to use more advanced XIDE features as well.

1.4 Other resources

There are several resources where more information about XIDE can be found:

- Evgenia Samochadina's Msc Thesis "XIDE - a visual component-based web application editor" <http://xformsdb-ide.googlecode.com/files/Msc%20Thesis%20draft.docx> contains detailed description of process of XIDE design, implementation and evaluation
- XIDE open source project page on <http://code.google.com/p/xformsdb-ide/> provides source codes and technical details about XIDE installation and deployment
- XFormsDB open source project on <http://code.google.com/p/xformsdb/> provides source codes and technical details about XFormsDB technology
- XIDE welcome page <http://testbed.tml.hut.fi/xide/> contains demo videos and information about demo applications

Questions and comments are highly welcome by email samochadina@gmail.com

1.5 Document overview

This manual can be divided into two parts. First part contains brief introduction to the XIDE and demonstrates how to create and publish the application. It is made in a form of step-by-step tutorial and leads the user through basic XIDE workflow. This part is designed for *non-professional* or novice *professional* users, who have never used XIDE before.

Second part describes advanced features and is supposed to be a reference book for *professional* users, who need a clue how to perform a specific action. Features are grouped by user tasks.

1.6 Legend

Bold text is used to highlight labels, buttons or links from the user interface.

Italic text is used to highlight important terms.

2 Getting started tutorial

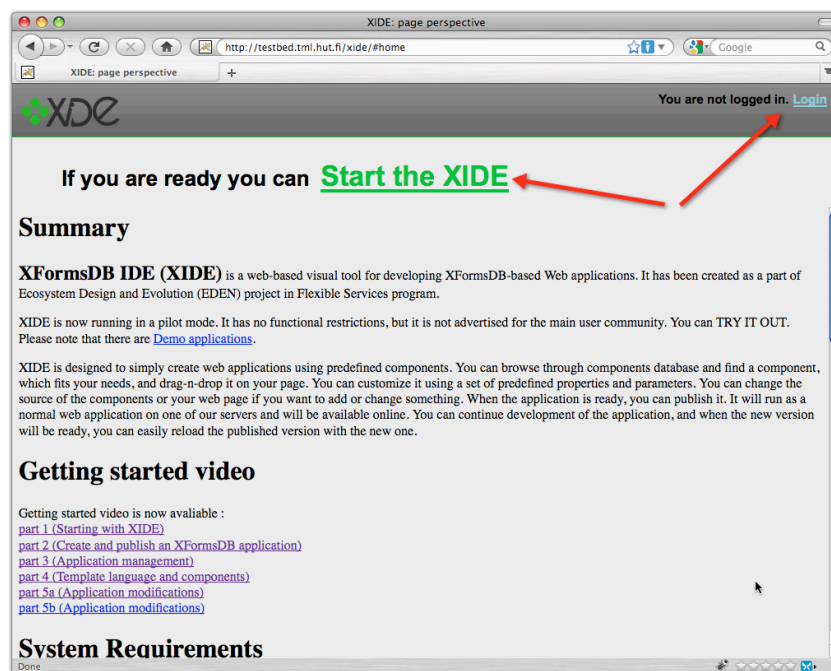
In this tutorial you will learn how to how to create and publish your first XFormsDB-based application in XIDE. It can be used as a brief introduction to XIDE and covers basic workflow for *non-professional* user. For more information, please see sources described in section 1.3 and advanced tutorials section.

This tutorial is a step-by-step instruction, which shows you how to create and publish the application in XIDE.

Step 1: XIDE welcome page

XIDE can be accessed on <http://testbed.tml.hut.fi/xide/>. This is a XIDE welcome page, which contains information about XIDE main ideas, system requirements, demo videos, demo applications descriptions and other important issues.

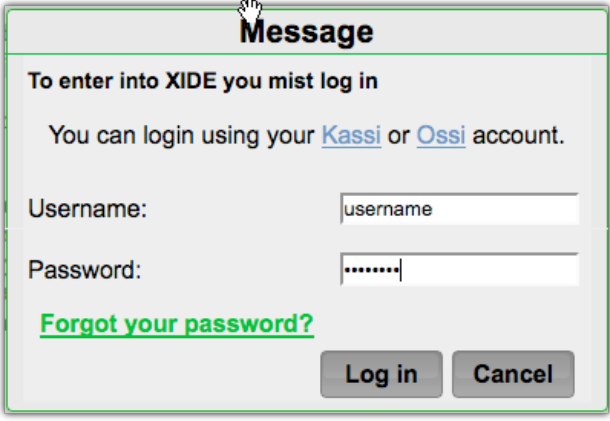
On this page you can also become familiar with XIDE layout. There is always a *header* on the top of the page and main content area. Header contains login information and different links, while main content area displays valuable information. There are 4 system screens or *views* in XIDE; each of them is designed to help user to perform specific task. While user works with XIDE, system switches between different views depending on the current task. More information about views can be found in section Step 1: 3.1.



To start the XIDE, you can click on **Start the XIDE** or **Log in** link.

Step 2: Log in to XIDE

In order to start using XIDE, you need to log in. If you don't have Kassi or Ossi account, you need to create one. Easy way to do it in English is to Ossi service.

A login message box titled "Message". It contains the text "To enter into XIDE you must log in" and "You can login using your [Kassi](#) or [Ossi](#) account." Below this are two input fields: "Username:" with a placeholder "username" and "Password:" with a placeholder ".....". A green link "[Forgot your password?](#)" is below the password field. At the bottom are two buttons: "Log in" and "Cancel".

Message

To enter into XIDE you must log in

You can login using your [Kassi](#) or [Ossi](#) account.

Username:

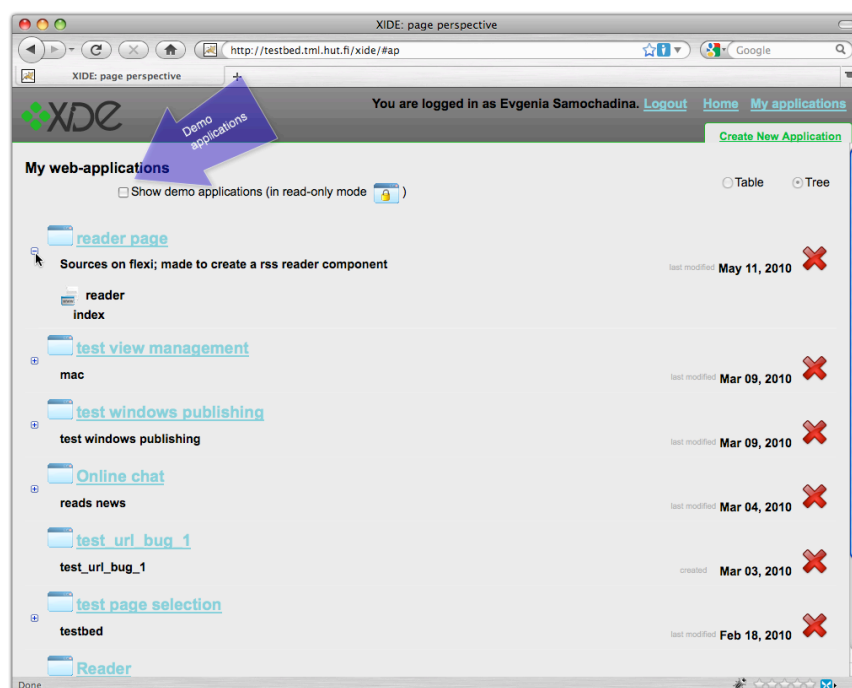
Password:

[Forgot your password?](#)

When you successfully logged in, XIDE is loaded.

Step 3: Application List View

Application List View is loaded. This view is used for reviewing existing applications and pages. Now you can see a list of applications you have an access to. If you are new to XIDE, there are no applications in the list, since you haven't created any yet.

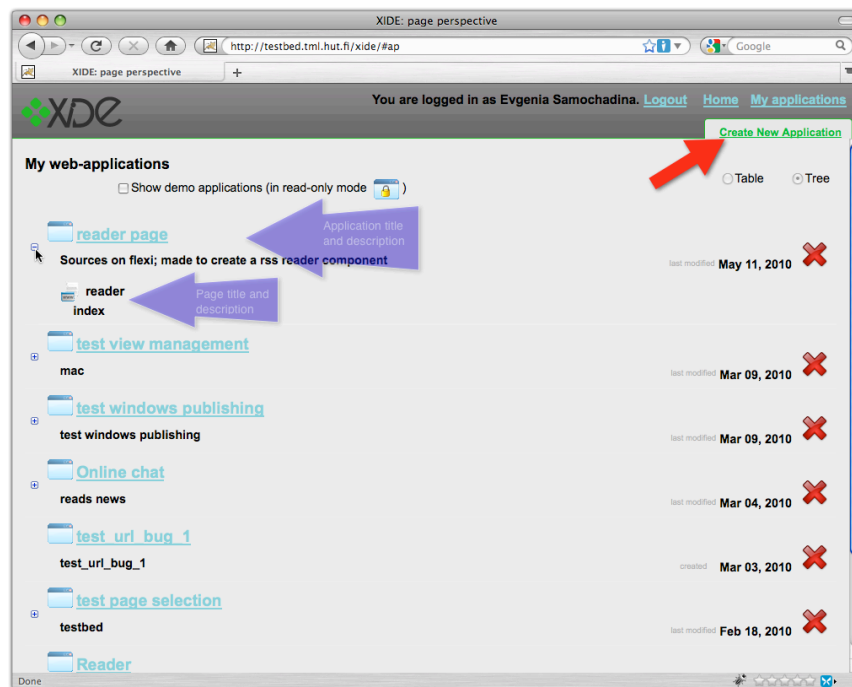


XIDE has demo applications, which are public and run in read-only mode. They are useful to see XIDE possibilities and check how things should be done. You can find information about demo applications on XIDE welcome page.

You can click on the **Show demo applications** link to see demo applications.

Step 4: Application List View: Applications and Pages

Now several applications appeared. In XIDE there are applications and pages. Application can contain one or more pages.



You can see each application has title, description, modified date and published details (if it is published). You can see application pages if you click plus sign.

To create an application **Create New Application** link should be clicked.

Step 5: Create New Application

To create an application you need to click **Create New Application** link and start the wizard.

A wizard window appears. It contains fields need to be filled in order to start new application creation.

The 'Create new application' wizard window has the following fields and options:

- New application's title:** Text field with 'Test application' entered.
- New application's description:** Text field with 'This is test application used for user manual' entered.
- New application's related url:** Text field with 'user_manual_test' entered.
- I want to create a page:** Check this if you want to create main page for this application right after the application will be created. (unchecked)
- I want this application to be visible to others:** Check this if you want this application be shown to other users when they will look for example and allow them to make a copy of it. (unchecked)

Buttons: OK, Cancel

You should enter a descriptive name, description, path (one word; how the application will be displayed in the URL) and press **OK** button. Values you entered will be verified by XIDE and application creation procedure will be started.

In case if some value does not fit, you will see corresponding message.

Create new application

This is a last step before creating new application will start. New application will appear in the list of your applications right after creation procedure will be completed. You can modify most of the values you enter in this form later and create as many pages as you want. Please fill in the following fields and press OK button

New application's title
This title will be displayed in the application list. Should not contain any special symbols (e.g., '&'). E.g. 'BuyBook Online shop'

New application's description
This description will be displayed in the application list and help you to understand what is this application for. Should not contain any special symbols (e.g., '&'). E.g. 'Online shop with paper and audio books'

New application's related url
One word sequence; how the application will be displayed in the url of the browser. Should contain only '_', numbers and letters. E.g. 'buybook'

I want to create a page
Check this if you want to create main page for this application right after the application will be created

I want this application to be visible to others
Check this if you want this application to be visible to other users when they will look for example and allow them to make a copy of it.

Field cannot contain spaces

OK Cancel

When application has successfully created, Application View is loaded.

Step 6: Application View

In *Application View* you can view and modify *application* and its *pages* details. On the left you can see an application and its pages. Currently there is no page. If you select element from the left, you can see its *properties* on the right on **Properties Tab**.

Some of them are editable (e.g. description and title), others in read-only mode (e.g. dates).

Files Tab shows file structure of the application.

XIDE: page perspective

http://testbed.tml.hut.fi/xide/#cap/app=349

You are logged in as Evgenia Samochadina. Logout Home My applications

Delete Reload Publish/Unpublish Export

Test application Application we've just created

Add new page

Properties Properties Tab

Is this application a demo application (shown to others, cannot be edited)

Application descr
Descr of the Application

Application is published
Application is published and can be accessed online

Application modification date
Modification date of the Application

Application author
Author of the Application

Application title
Title of the Application

Application publishing date
publishing date of the Application

Application creation date
Creation date of the Application

Is public application
Is public Application (shown to others)

Application url
url of the Application

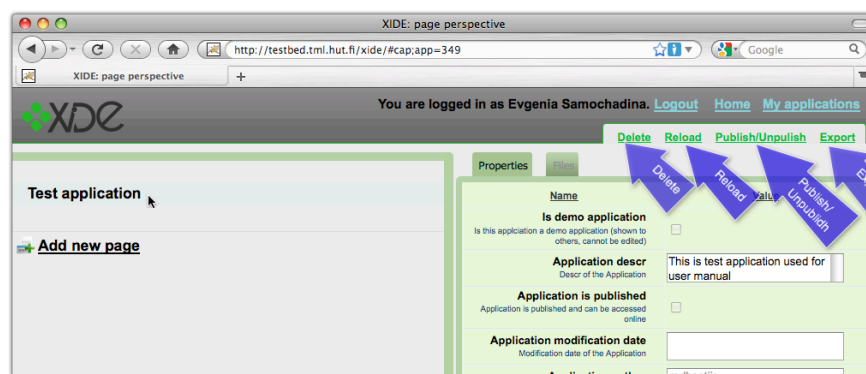
Related URL for application
Related URL for the application

Main page
Main page of the application. Is loaded as a welcome page when you type application URL.

Save Cancel

The folder structure is fixed, but you can add/remove additional files (e.g. CSS, images or queries). Please see more information about XFormsDB Applications in section 3.3.

In this view you can add and remove pages and manage application status: publish, stop, reload, delete. More information about application status management can be found in section 3.4.



To create new page you need to click on **Add New Page** link.

Step 7: Add New Page

To create an application you need to click **Add New Page** link and start the wizard.

A wizard window appears. It contains fields need to be filled in order to start new application creation.

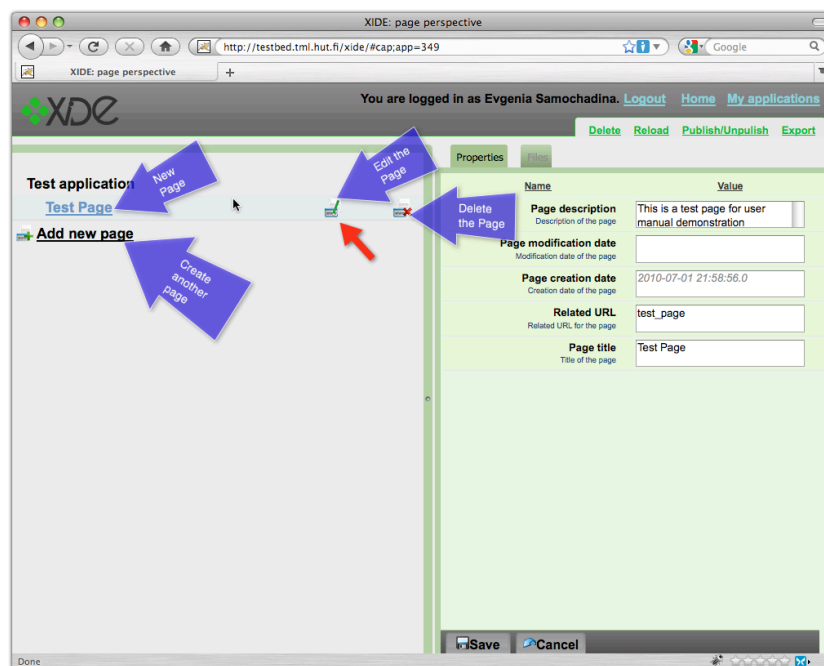
You should enter a descriptive name, description, path (one word; how the application will be displayed in the URL) and leave all other settings as default and press **OK** button. Values you entered will be verified by XIDE and new page creation procedure will be started.

In case if some value does not fit, you will see corresponding message.

As a result, XIDE will create a ready-to-use page. It will contain basic page structure and a default container for components, so you will be able to start adding components right away.

Step 8: Application View: Page management

When the page has successfully created, it appears in the Application View. You can now select it.



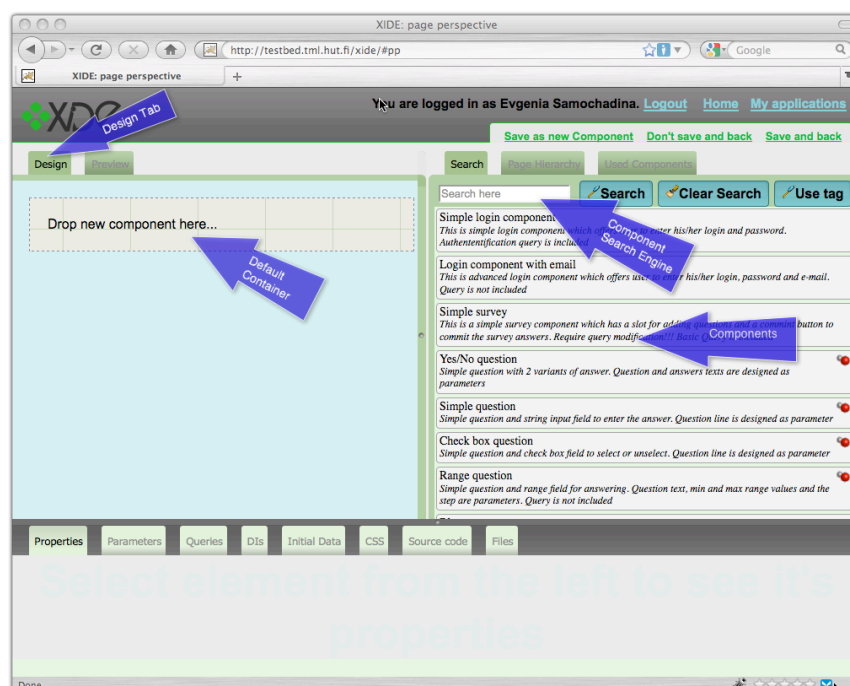
After the page has been selected, it is possible to modify its properties and edit or delete it.

To start editing you need to click on **Edit** icon.

Step 9: Page View

To start editing you need to click on corresponding **Edit** icon from the *Application View*. After that, *Page View* for selected page will be loaded.

In *Page View* you can view and edit page source. There are many tabs for different purposes. You will see their purpose later.



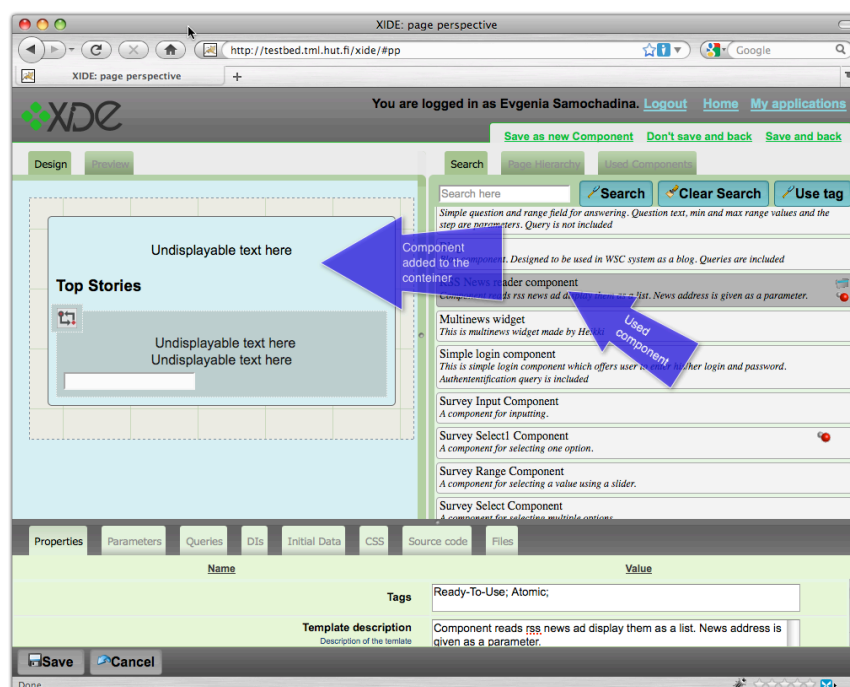
Design tab represents the page UI in a schematic manner. It does not display final outlook of the page; it shows the structure and main UI elements.

Our page has only single empty container created by default.

Search tab allows you to search for different templates, which can be drag-n-dropped into your page. It has a text and tag search, e.g. **ready-to-use** tag is quite useful – you can find the components, which do not require any additional settings.

Step 10: Page View: add a component

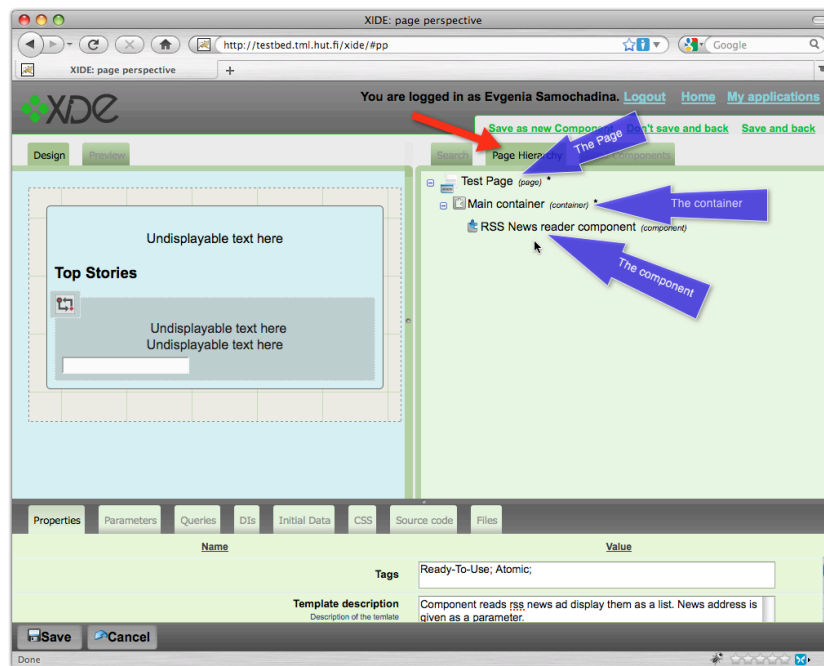
You can drag-n-drop component into container on the page. Container will highlight the place where you need to drop component. Let's select **RSS Reader** component and add it.



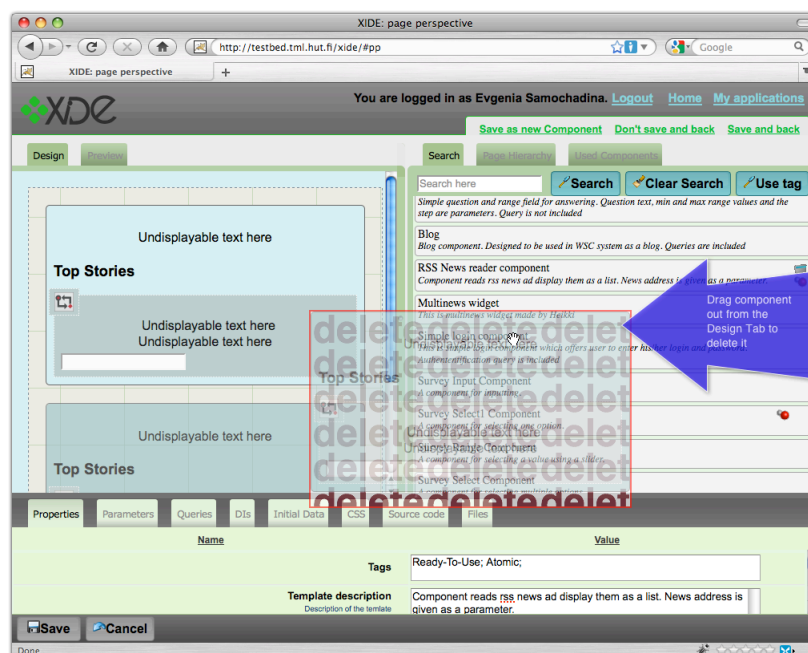
You can have as many components of the selected type as we want.

Step 11: Page View: change order of components

Click on the **Navigation** tab, which represents the page structure as a tree and allows you to manage containing elements right in this tree by drag-n-drop.

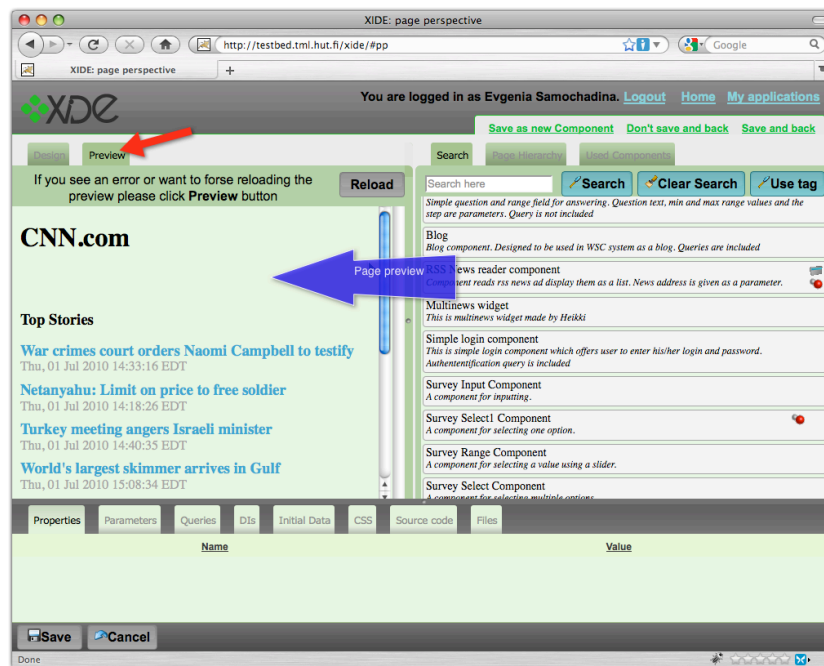


You can change the order or delete components by drag-n-drop on Design Tab.



Step 12: Page View: preview page

You can select **Preview** tab to see how the page will look like, when the application will be published.



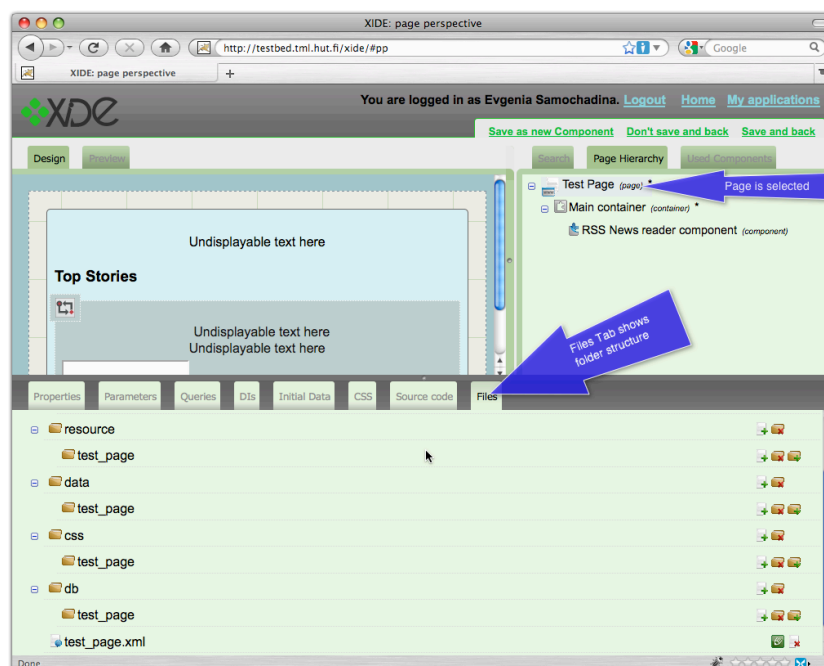
Step 13: Page View: view component's information

Bottom tab shows information related to the element, which is currently selected. You can select any component, container or the page from Design tab or Navigation tab, or component from Search Tab and see it's properties and other related information.

Properties tab shows different properties, e.g. title, description, tags.

Parameters tab shows parameters. By changing the parameters, you can influence on component's appearance and behavior.

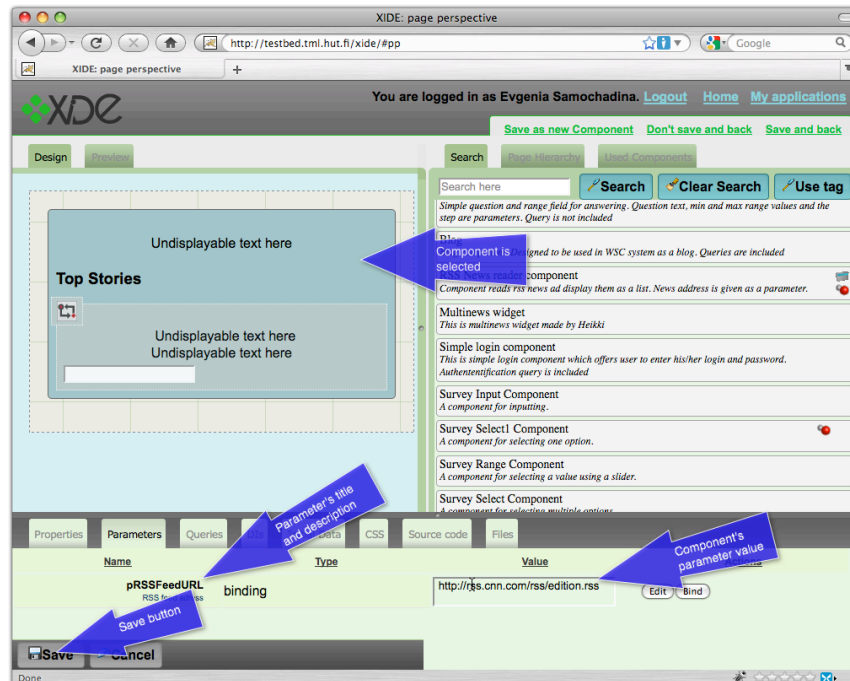
Files tab represents file structure of the selected element.



CSS, Queries, Data Instances, Initial Data tabs shows different types of external files that can be used by XFormsDB web page.

Step 14: Page View: customize component

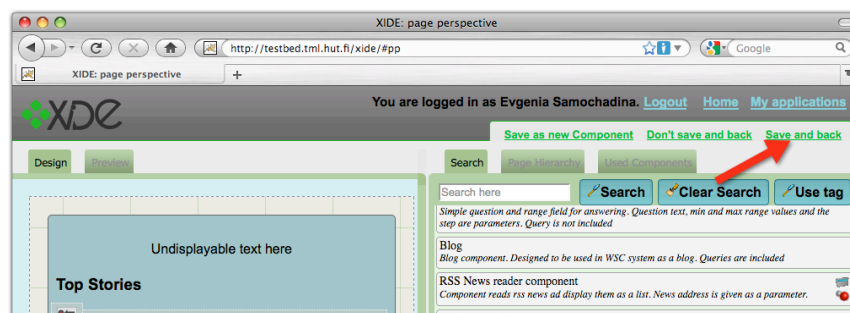
If you select the RSS component you've added previously and click on the **Parameters** tab, you can see that it has one parameter. This parameter contains the URL, which is used to collect the news.



To change the URL change the address and click **Save** button. To see that the component has changed, click on **Preview** tab.

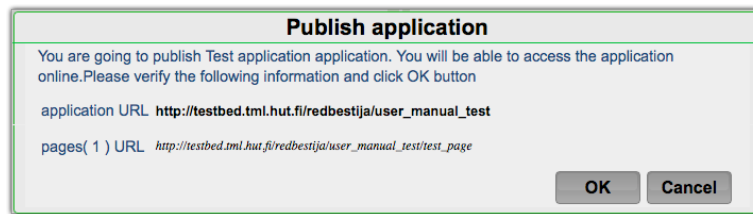
Step 15: Page View: save changes

When you are ready with the page, you can save it and publish the application. Click **Save and Back** button to save changes and return to the Application View.



Step 16: Application View: Publish the application

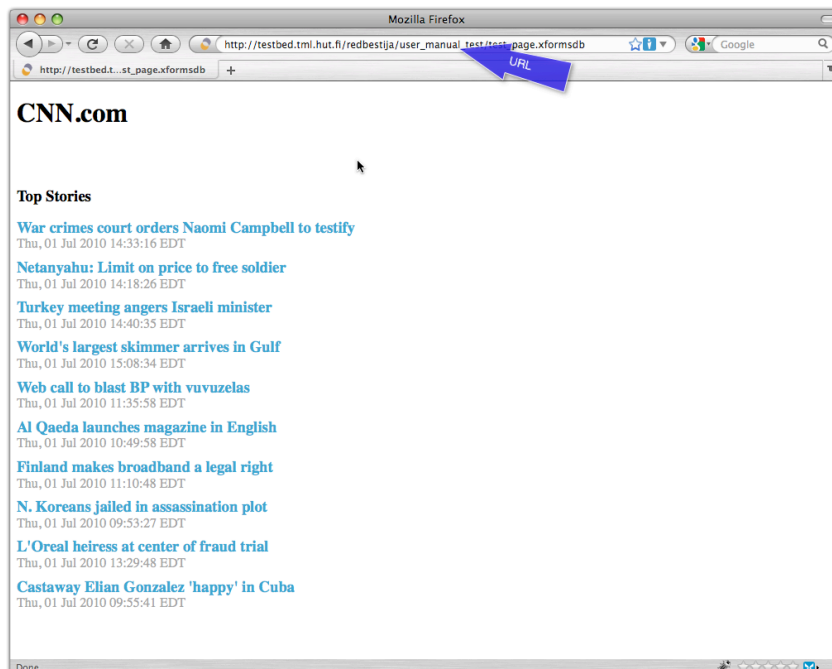
Click on the **Publish** button to start *Publish* dialog. Click OK button and wait for application to be published.



When the success message appears, you see the link where your application is published.



You can click on the given link and see the application published online.



Finally, you have created and published the application.

3 Advanced tutorials

This section contains advanced tutorials about different features of XIDE. Please, refer to the table of contents to find, which tutorial contains information you are interested in.

Page View: change page's source code

Source code tab shows source code of the selected element (component or page). You can easily modify the page source code to fit your needs.

Let's add a bold text before the container.

3.1 View management

XIDE has several views (or screens), which are designed to be used in different user tasks. When user works in XIDE, he/she switches between views depending on the current task.

XIDE should have 4 views: Welcome view, Application List View, Application View and Page View. Possible transitions between views are displayed on Figure 1.

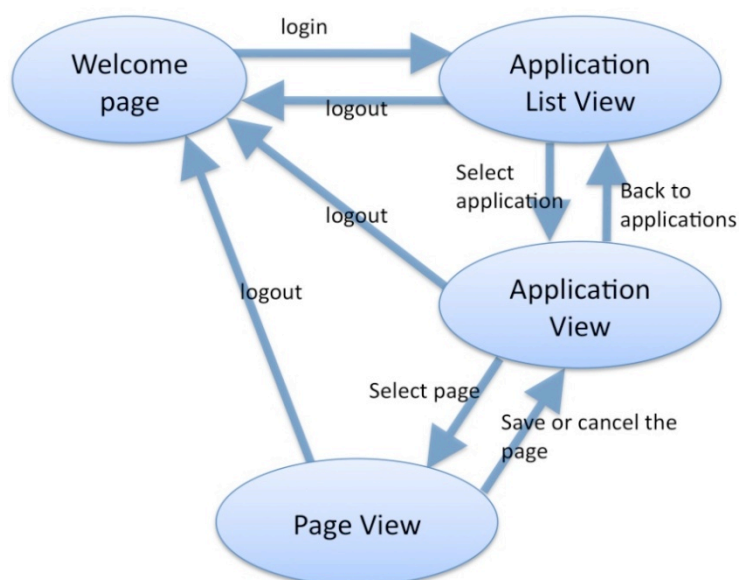


Figure 1 XIDE view management

On the *Welcome Page* user can find information about XIDE main ideas, demo application descriptions, help videos, system requirements and other important issues.

Application List View provides list of applications user can access. User can see application pages if he/she clicks plus sign. This view is used to find the application user wants to modify or to create new application.

If user selects the application, *Application View* appears. It is possible to view and modify application and its pages details here, manage the application (publish, stop, reload the application; delete and create a page).

Page View is loaded when user wants to edit a page. It has several tabs and many features to help user in page design and implementation. It is possible to manage page content both visually and by changing the source code and preview the page.

3.2 XIDE concepts

3.2.1 Application

Some of the system concepts are almost obvious. So, *Application* is a web site - a set of Web Pages developed by the user that represents some functionality (home page, internet shop...). Creating of the application mostly consist of creating of web pages it is formed of. When the application is ready, it can become available online. Additionally, it can be exported as a service to be used in some service composer tool.

3.2.2 Web Page

Web Page is an entire page, one of the application's pages. Web page has a header section that represents properties specific for the web pages. Web page can have external files, defined for plain XFormsDB page (CSS, data instances, queries, resources). Web page source code (main XML file) is formed from plain XFormsDB code and reusable elements.

3.2.3 Component

A *Component* is an element, which represents a piece of functionality. It is made based on XFormsDB technology. There can be components with different complexity. Since component is customizable, it should have properties and parameters, which can change its appearance or behavior or set up a link between two components. It is reusable.

Component is defined once in the system's database, but it can be called from any web page. In addition to component's source code (XFormsDB), it can have its own external files, which can be used by XFormsDB application (CSS, queries, data instances). Finally, component object can have meta-information

(component general information and description, tags, settings, parameters), component's source code and other files.

3.2.4 Container

Container is a place where components can be positioned on the page. Container has graphical interface so it is easy to see where components can be added. Container can have a embedded validation mechanism, so it can check whether the component can be added inside this container. Additionally, container is a grouping element, which can share files (data instances, queries and CSS) between its children components and manage group access rights. File sharing means that the children components can refer to container's file in their source code. Access rights can be set to container and after that they will be automatically propagated to the children components. Containers are hardcoded in the page source code.

3.2.5 Template Language

Visual representation of the concepts can be found on the diagram (see Figure 2). A web page has containers for components (Container 1 and Container 2) and plain XformsDB parts. Each container can have components inside. Each of Web Page, Container and Component can have its own external files, which are shared between its children.

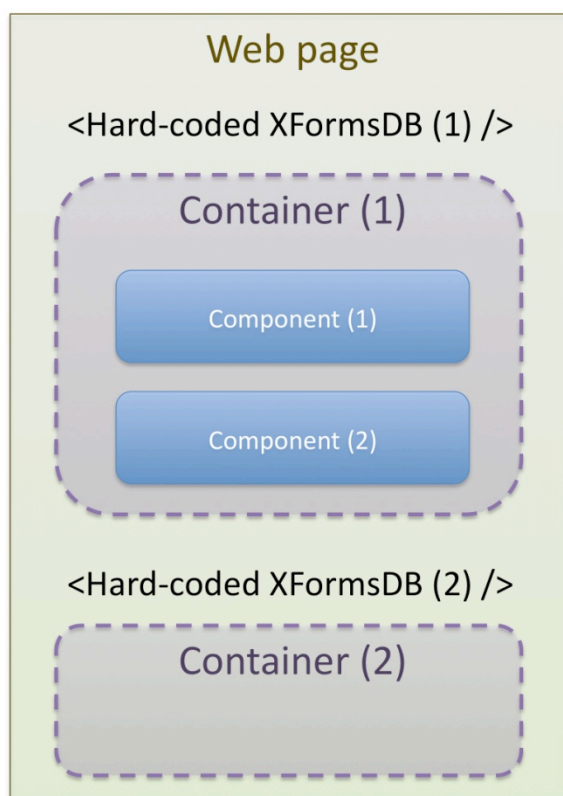


Figure 2 Concept of the Web Page

Finally, there are three main concepts: Web Page, Container and Component. These objects exist only in XIDE, and user handles them during the application creation. When the application is ready and user wants to publish it, these objects should be transformed to plain XFormsDB application.

Intermediate declarative language is used in order to support using of components and containers. Template Language (TL) is XML based markup language. Please, see Template Language API documentation on XODE project page (http://xformsdb-ide.googlecode.com/files/The_API_Specification_for_Template.doc). XIDE demo applications can be referred as a examples of template language using.

3.3 XFormsDB application

3.3.1 What is XFormsDB?

Applications, created in XIDE, are XFormsDB applications. For more information about the technology please check <http://code.google.com/p/xformsdb/>.

XFormsDB is an extension of XForms, W3C standardized technology for creating interactive web forms. With XFormsDB it is possible to create a web application by writing only client-side code using declarative languages. XFormsDB is claimed to be useful for authoring highly interactive multi-user web applications by non-professionals.

XFormsDB application consists of one or more web pages. A web page is a page written on XFormsDB language. Talking in a nutshell, XFormsDB web page is a XML-based document, where XForms, XFormsDB and HTML tags are combined.

3.3.2 XFormsDB application structure

Authoring XFormsDB web pages involves using different declarative technologies for different purposes (see Figure 3):

- XHTML for document structure
- XFormsDB for data access and common server-side tasks and XForms for user interaction
- XML for data modeling and interchange
- CSS for visual layout and presentation
- XQuery and XPath for querying data
- Different external Resources, e.g. images or JavaScript, that can be built into XHTML

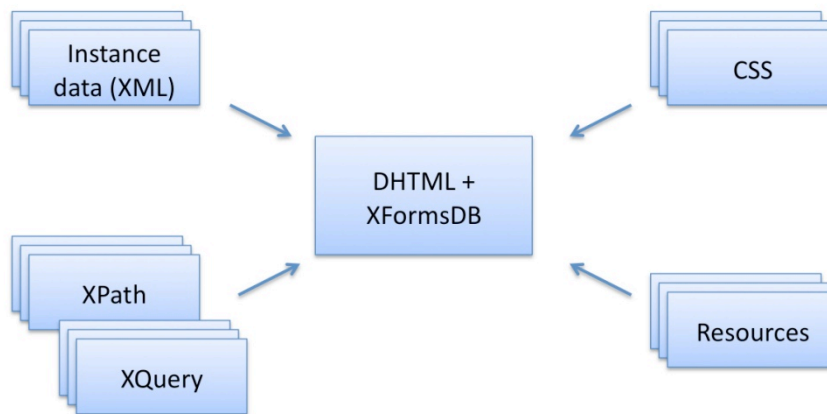


Figure 3 XFormsDB web page components

Practically each web page has a source file, which defines page structure and user interface. Additionally, there can be different components introduced, e.g. instance data, XPath and XQuery equations, CSS, external resources. Generally, each component can be either described in external file or specified directly in the source code. Using external files provides reusability and reduces complexity of the page source code. However, in case of complex application there can be a mess of files.

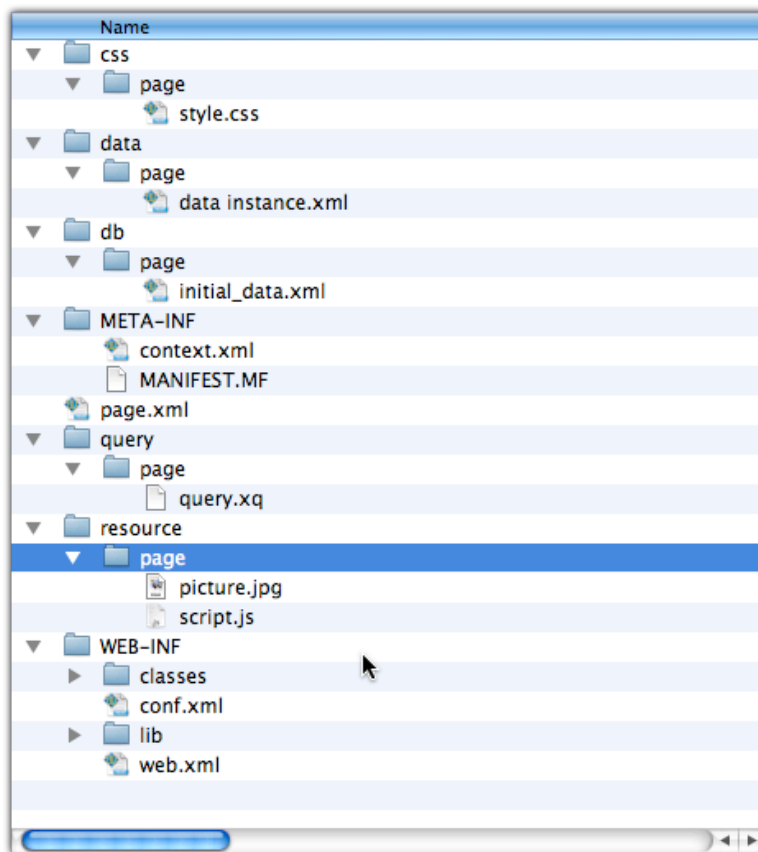
Except external component files, XFormsDB application also contains secondary files and folders used when the application is running. Several obligatory configuration files should be set up. They cover different technical issues, for example, how the application should be deployed on the server or how to connect to the database.

3.3.3 XFormsDB application in XIDE

In XIDE web page is formatted with markup language, called Template Language (TL), designed especially for XIDE. It is used for reusable components description and management: adding them to the page and configuring. To describe briefly, TL defines the web page, components, and placeholders for the components. When the application is published, it is transformed to plain XFormsDB.

3.3.4 Application Folder structure

In XIDE application folder structure hierarchy is fixed. It has folders for each type of files. In those folders, files are also grouped by the pages they belong to. However folder structure cannot be changed, new resource files can be added. The only exception is /db folder. This folder is responsible for data structure in the database, so it should be possible to create a folder structure under it.



3.3.5 File access rights and sharing

There is file access propagation in the application folder structure. If the file is under /resource folder, it is shared between all application pages. If the file is under /resource/page_id folder, it can be accessed only by specified page. This should be noted when new file is uploaded.

3.4 Published application management

Other feature to pay attention to is management of the published application. There are different statuses of the application (see Figure 4), depending on XML database management and visibility of the application.

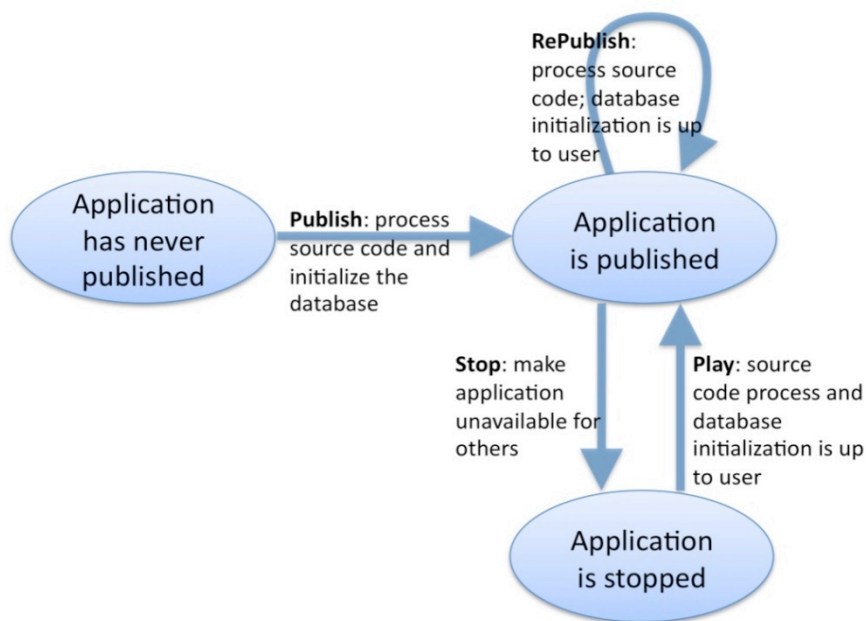


Figure 4 Published application Management

There are two main parts in the application publishing process. First, application source code is transformed from Template Language to XFormsDB and second is database initialization.

When an application is *published* for the first time, its source code is transformed and the data is loaded into the database.

After that, a user can *stop* the application, which means it should not be visible to other users. Later on, he/she may want to make it available for others (*play*) again, without repeating the publishing procedure. In case if application source code has changed, a user should be asked if he/she wants to play the old version or publish a new one and should the system reset the database of the application.

Republish option is useful, if the application is currently running, but a user wants to update it with the new version.