WebUI Get Started Guide

This document is meant to accompany a live AIMMS WebUI demo that will be given (either live or online via a GoToMeeting session). After attending the demo, this document should provide you with the initial steps to get started and play around with the AIMMS WebUI yourself. Support is available to you during the beta testing of the AIMMS WebUI.

# Requirements

To be able to **develop** a web interface for your AIMMS application using the AIMMS WebUI beta, you need a web-enabled version of AIMMS plus some dedicated AIMMS WebUI components. The web-enabled AIMMS version is just a normal AIMMS 4.1 version and can be installed alongside any existing AIMMS 3.14 (or AIMMS 3.13, …) version. In addition, you will need to have the [Chrome](https://www.google.com/intl/en/chrome/browser/) browser installed in your machine. Eventually, the AIMMS WebUI will support other HTML5 browsers too, but for now, the beta is targeted towards Chrome.

To install the AIMMS WebUI, download the latest version of the AIMMS installer and the WebUI installer from <ftp://ftp.aimms.com/pub/Download/Private/WebUI_43827>. Please update *both* products (AIMMS and the AIMMS WebUI) if you are upgrading an existing installation.

# Deployment

The AIMMS PRO installation (PRO 2.0, to be released) is extended with WebUI support too. There is no need to install the WebUI on your PRO server separately. In the end, it will just be a matter of publishing a WebUI enabled version of your AIMMS application. Once you launch such an application from the PRO portal, a local browser session will be started that shows the WebUI of your application. The integration with the PRO framework is not yet part of the beta-test program.

# The WebUI Components

The WebUI installer installs several files in the C:\Program Files (x86)\AIMMS\WebUI folder. These files consist of

* The Cube Server: a collection of DLLs, responsible of communicating data between AIMMS and the WebUI (available in the Win32 and x64 subfolders)
* The WebUI Server: a java server responsible of storing (and retrieving) the pages and widgets that are developed
* The AIMMS WebUI runtime: a collection of JavaScript and other resource files
* Some example AIMMS projects (for which some example WebUI pages have been developed)
* A version of the Java Runtime Library (JRE7)
* Some logging configuration files that allow you to control the destination and amount of logging information that is created by the Cube Server and WebUI Server

Assuming that the WebUI has been installed properly, once you open an AIMMS project, the **Tools** menu will be extended with a **Start Web UI** and a **Stop Web UI** command. Starting the WebUI will start the Cube Server and launch a WebUI Server on your local machine. By default, the Cube Server serves on port 12002 and the WebUI Server serves on port 12001. Amongst others, the WebUI Server serves files with C:\Program Files (x86)\AIMMS\WebUI\www as the document root. In addition, the WebUI Server stores widget content in the *WebUI* subfolder of your AIMMS project.

# The WebUI Application folder layout

An AIMMS WebUI-ready application is an ordinary AIMMS application that contains a *WebUI* subfolder of the project folder. All WebUI pages and widgets will be stored (by the WebUI Server) in the *pages* and *widgets* subfolder of the *WebUI* folder. In addition, it is possible to store application specific *resources* in the resources subfolder (of the *WebUI* folder). Application specific images should be stored in the *resources/images* subfolder. Application specific JavaScript files (e.g. widget (addons)) and CSS (styling) should be stored in the *resources/javascript* and *resources/css* subfolders resp.

# Opening an Existing AIMMS WebUI Example Page

First, start the web-enabled AIMMS version you installed and unzip and open the **MealsRUsWebApp** example that can be found in C:\Program Files x86)\AIMMS\WebUI\Examples\MealsRUsWebApp.zip. Whenever you upgrade to a new version of the WebUI, make sure to also open the latest version of the example projects (to make sure that any updates in the example AIMMS projects are propagated correctly to the WebUI). Once the project is opened, start the WebUI by pressing the **Start Web UI** command from the **Tools** menu. A command window will open that shows the output of the WebUI Server that is running. Finally, open the (Chrome) browser and navigate to <http://localhost:12001/MealsRUs>.

# Creating a Your Own AIMMS WebUI Page

To create a new page for your model, just press the hamburger icon on the top left position of your browser window. A navigation menu will open. Just, press the big + at the bottom of this window, specify a name for your new page and press the + button to add the new page to the list of available pages. To navigate to the newly created pages, just press the page in the list of available pages. Please note, that you are free to use slashes in the name of you page in order to categorize the pages for you application.

In order to create new widget on your page, just press the **Edit Page** button, the ‘pencil’ icon next to the page title. A widget panel will fly out from the left. This widget panel lists all widgets that are present on the page. By pressing the big + at the bottom of this list, a New Widget wizard will popup that allows you to specify contents, name and widget-type (e.g. table, bar chart). Since the widget database is shared for all pages for your application, please make sure that the names for all pages and widgets are unique throughout your application.

The ‘cog’ button (in the upper right corner of a widget) will open a popup window that allows you to change the options for the widget. Option changes are automatically synchronized with the WebUI Server.

# Data Management Support

To be able to use the data management features through the WebUI, the ‘Data management style’ option of your AIMMS application has to be set to ‘Disk files and folder’. This is however the default for new AIMMS 4 projects. The WebUI has been extended with a data management pane, which lists all available data cases. The data management pane allows you to load, save and delete cases associated to your AIMMS project.

# Extending your AIMMS Model with support for the WebUI

To be able to control what identifiers are visible to the WebUI (e.g. when selecting the contents for a widget), just extend your AIMMS model with a set AllPublicIdentifiers (in the global namespace), subset of the set AllIdentifiers and to be initialized with the identifiers that are public to the WebUI. Currently, if the set AllPublicIdentifiers has not been defined, all identifiers in you model are available to the WebUI. However, it is possible that this default behavior might change in the near future.

The legend widget that is available in the WebUI provides the possibility to add, delete and rename set elements in your model. Consider a set S in your model. Whenever your AIMMS model contains a procedure uponAddElement\_S, uponRemoveElement\_S or uponRenameElement\_S resp., the procedure will be called when the corresponding set is manipulated triggered by the WebUI. For example, the uponAddElement procedure can be used to initialize model data for the newly created element.

# Extending the WebUI Framework with your own Widgets

The WebUI is built upon open framework called AWF (AIMMS WebUI Framework). AWF is a very open framework that allows you to add you own widgets. However, creating your own widgets requires (advanced) knowledge of the AIMMS Widget Framework (AWF), HTML5/JavaScript/CSS and the Cube Server data interface. At this very moment we did not publish the API of this framework. However, if you want to play around with developing you own widgets, you are advised to take a closer look at the code for a sample [simple-table-widget](ftp://ftp.aimms.com/pub/Download/Private/WebUI_43827/simple-table-widget.zip) that can be downloaded from the ftp site mentioned in the introduction of this document.

# Change the Styling of Web App through CSS

It is possible to (re)style your web application by providing a custom CSS. The example page shows a small drop down button in the upper right corner of the page that lets you switch between several example styles. Please note, that the class names that are referred to in the CSS might change in the future. Application specific CSS files should be stores in the *resources/css* subfolder of the *WebUI* subfolder of your project folder. As an example of application specific styling the **MealsRUsWebApp** has been extended with a *theme-switch-addon* that consist of some JavaScript and CSS that result in the theme switch drop-down button being shown in the upper right corner of your web application.

# Multilanguage Support

The AWF runtime library has been extended with multi-language support. Depending on the language settings of your browser, all strings that are displayed in the WebUI will be checked against a language specific translation table. If a translation is available, the translation is displayed. Otherwise, the original string is displayed. These language translation tables can be found in the *C:\Program Files (x86)\AIMMS\WebUI\www\resources\languages* folder on your computer. Please not that, when making changes to a language file, it might be necessary to startup your text editor in *Administrator Mode* to prevent all kinds of Virtual Store related problems. Eventually, we intend to support the storage of application specific language files in the *resources/languages* subfolder of the *WebUI* subfolder of your project folder.

# Recent/Current/Upcoming Developments

We are currently in the process of finalizing the integration with the AIMMS PRO framework.

Since the provided WebUI version is still in a beta stage, please note, that it might occur that some things (intended or not) stop functioning when you upgrade a newer version.

# Questions/Issues

If I have not already done so, I will try to give a small introductorily demo (either in person or online via a GoToMeeting session). Please contact me by email, Skype or phone whenever you have question about the AIMMS WebUI.

Koos Heerink

AIMMS B.V. – Haarlem – The Netherlands

Email : [Koos.Heerink@aimms.com](mailto:Koos.Heerink@aimms.com)

Skype Name : koos.heerink

Phone : +31 (0) 23 5511512