Iguana – General

1 Introduction

This document contains background information and describes Iguana's main features and technical concepts. It also contains information on how to get started and on how to log on to the Iguana CMS as well as general information on the structure of the user interface.

1.1 What is Iguana?

Iguana is a marketing, communication and personalization platform that allows you to promote content of your organisation on the web, via websites, mobile interfaces and technologies such as email or RSS. As such Iguana is a marketing tool, that has sophisticated personalization features and that allows you to expose your content and services wherever the customer is.

Technically, the main components of Iguana are its front-end framework, it's service layer (called IguanaXgateway), a set of server functions, mobile interfaces and a CMS (Content Management System) that supports these features. The front-end framework, which is widget based, presents content in widget containers which can be moved around and placed anywhere on a page. IguanaXgateway is the service layer through which the front-end widgets retrieve their content from back-office applications. Iguana cannot only expose content on a website, but also via mobile interfaces, publish services such as RSS and traditional email messaging. The IguanaXgateway can also expose content and services via a programmatic interface (usually XML).

The configuration of these elements is done via Iguana's CMS.

1.1.1 Iguana features

This section lists, at a very high level, the main Iguana features. The section contains no "hard" factual information, but describes the main benefits of using Iguana.

Iguana: a modern approach to services

Iguana is a new way to promote your library's services. It is an online platform that brings together your collection and services. Iguana uses web access, mobile interfaces and push technologies such as interest emails and RSS feeds to expose both collections and services. It is an environment where your users learn about your services, explore your collections and personalize their experience.

A single environment

Most libraries have a web site and an online catalogue that often exist next to each other as separate entities. With Iguana, we merge both environments. This approach provides you with the benefit of promoting both your services and collections in one place.

Personalized and dynamic

Iguana focuses on mobility and personalization. This means that your online environment is no longer static. It is one in which your users view information that is tailored to their interests. Users can also decide what content to view and take information along to other websites such as iGoogle. Your users enjoy direct access to their user activities and can discover information in your catalogue and beyond.

Functional

Iguana is rich in features both visually and functionally. The platform delivers a true Web 2.0 experience by focusing on the user and placing the full breadth of your services front-and-centre. The Iguana platform provides the library with a flexible web presence. You can create as many pages as you want. Each page can highlight a particular part of your collection - say music or fiction - or a particular service such as this week's library events. With Iguana, you can provide a different 'look & feel' for the various audiences that you serve. Think of a kids' theme or a theme for the visually impaired.

All themes and pages are based on an underlying structure that emphasizes easy access to information and user activities. Every page can include a search box and can feature a 'My Library' button, which provides users immediate access to their library account.

Widgets

Iguana is widget-based. The widget technology enables you to configure your site in a very flexible way.

Featured items

A key feature is the presentation of 'featured items' throughout your site. Featured items are covers of books, DVDs, CDs, etc. that are displayed in an animation. When hovering over the cover (or clicking on it in some interfaces), the user views the item's details and can immediately place a reservation. The featured items Flash animation can be placed on any page - either on top of or within the main body of the page. You can have separate animations for different material types. In this way you can showcase music, digital books, large print books, DVDs, events in the library, and more.

Animations can also be personalized for each user. Within Iguana the user can select from a list of interest topics. After logging in, the animation will display items that have been tailored towards the user's specific interests.

Mashups

Through mashups you can integrate different types of content within Iguana. You can overlay your hours and locations with Google maps, for example. You can also include your library's Twitter feed and display this prominently on your home page.

You can also integrate an RSS feed from any source alongside related content. If you have a page for bestsellers, just present an RSS feed from your newspaper's bestseller list.

Discovery

Iguana provides your users with a rich discovery environment. Users can initiate a search on any page, and search both your site and catalogue at the same time. Search results display the item's cover including the full record details. Users can immediately place reservations and share items with others in external environments. They can sort results, view associated works, and restrict their search further through facets (e.g. location, material type and language).

The library may also include links to external content through a Wikipedia word cloud or via the integration of an OpenURL resolver.

My Profile

My Profile is the user profile component of Iguana. Here users can link their account to other users (say a parent who wants to view a child's account), view their loan history and place reservations. They can also create a personal interest profile with their favourite authors or genres. Iguana automatically collects new titles that match the user's interest profile and displays these in a personalized animation. Users can also register to receive updates of newly available titles via email.

Catalogue links

Iguana enables you to link any content on a page to content within your catalogue. Let's say that you have an event about Scandinavian thrillers. You can highlight the event on your events page and include direct links to related search results in your catalogue. These direct links are not hyperlinks to results on a separate page, but are records from the catalogue incorporated in your events page (in our example).

1.2 Main technical concepts

1.2.1 Iguana's technical design

Technically, Iguana is a front-end widget platform that links through a service layer (called IguanaXgateway) to back-office applications. The platform relies heavily on AJAX technology (Asynchronous JavaScript and XML, - see http://en.wikipedia.org/wiki/Ajax_(programming) for a high level introduction), one of the key components of Web 2.0 technology.

The implication of taking this dynamic, AJAX-based approach is that the web front-end is more of an environment in which applications can live and communicate. One can interact with these applications seamlessly through a more community-driven approach. This approach is very different from a traditional website approach where one moves from page to page, 'losing' each page as a new one is opened, with a heavy focus on exposing one functionality at a time. In short, the difference is that an Iguana 'website' is a container of application-states where states can change without affecting the container (no page-switching necessary), whereas in a traditional approach the website is the state, so a state-change means the page needs to refresh/change.

Some of the major advantages of the Iguana framework are:

- It can offer a much more dynamic experience because a change inside an application does not mean the entire page needs to be reloaded.
- Applications are much more independent from one another, meaning it will be easier to develop and integrate applications.
- The framework can and does supply reusable elements and functionalities that the applications can use.
- If an application creates an error, it doesn't take the whole framework down with it.
- Not only the look of the site can be changed by the customer, but also the feel, with arranging applications (gadgets, widgets) and adding and removing them as needed.

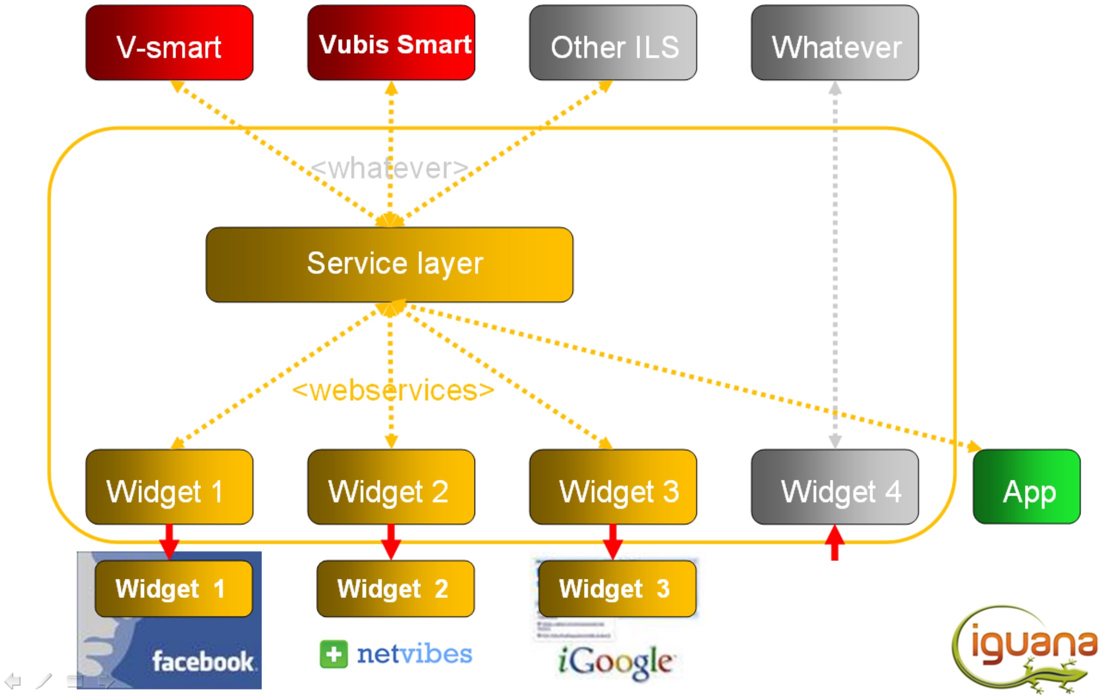
1.2.2 Main components

Iguana consists of the following technical components:

- a front-end widget platform that supports both "Iguana widgets" and third party widgets
- front-end widgets (called applications), that "live" in the widget platform
- a service layer called IguanaXgateway: this service layer takes care of the communication between the front-end widgets and the back-office applications; IguanaXgateway can also be used by third-party applications to get access to back-end data and services
- server-side components such as PHP.

The back-office applications are not part of Iguana, although Iguana does provide a number of interfaces (services) to these applications.

The following chart shows a simplified overview of the Iguana structure, in which Iguana is represented by the orange rectangle.



1.2.3 Front-end widget platform

The framework is the skeleton that binds the applications together and ties it all up into one single ‘website’. It comprises of several elements, some of which live client-side, some server-side and some provide communication between client and server. The framework is package- and class/object-based both client-side and server-side. This means that functionality is separated as much as possible (but not so much that it becomes too fine-grained), which also makes it easier to remove dependencies between objects.

The framework is the glue that holds everything together. It also provides common functionalities that can be accessed by all applications.

The basic flow is as follows:

- an application registers itself with the framework
- the framework checks the settings and parameters of the application
- if the application needs to be shown the framework creates a container for the application
- the container is inserted into the template “section” (often a column) it needs to be put in
- the framework calls the application’s start-up method, if present.

Communication between applications is asynchronous. That is, one application cannot directly catch the result of an action inside another application.

Another consequence is that an application can publish an event and no, one or several applications can pick this up and act on the event. The originating application has no way of knowing who is listening, nor do the registered applications have an idea who published the event, although the originating application can send this information along with the event if necessary.

An application can be placed in the framework in different ways. Typically, most applications will live within the framework directly and use Iguana's client-side JavaScript. Another method is to include an application in an iFrame. This can be a local application or an application from another domain.

When the framework gets opened inside a browser, a series of events is fired off before anything is displayed. This is done to determine what to display and how, as this might depend on the browser/reader used. This is done by checking the user-agent, which is either set from a previous visit, or is determined otherwise.

The user-agent is then used to determine both the initial HTML that is shown and the settings, files and applications that are loaded in from the initial HTML.

After the start-up details are known, the framework displays the initial starting/loading screen and queries the server for the list of applications, which files to load for the rest of the framework, and so on. At this point the HTML is small and only a couple of additional files are loaded (basic CSS and basic Iguana core JavaScript).

The dojo JavaScript framework

Iguana uses a JavaScript framework called dojo (see http://www.dojotoolkit.org/). Dojo is an extensive toolkit with support for lots of widget types. Dojo also has good accessibility / WAI-ARIA support, which is built-in into its widgets.

1.2.4 IguanaXgateway

The IguanaXgateway is the service layer for Iguana; this service layer is an integral part of the Iguana product. The IguanaXgateway provides access to a set of services via either SOAP or HTTP/URL and does not contain any user interface (visual) elements.

The list of services that are provided as part of the IguanaXgateway can be found in the document *IguanaXgateway _ available services _ v2.xlsx*.

Customers who want to use IguanaXgateway services can get the technical documentation provided they confirm to the usage restrictions and sign an NDA.

If customers want to deploy IguanaXgateway services and require Infor to assist in the implementation, the consultancy will be charged as PS on a "work delivered" basis, or via a "prepaid" construction (e.g. two days of consultancy are provided; additional hours are charged on a "work delivered" basis).

1.2.5 Technologies used

Iguana uses multiple technologies. The most important ones are:

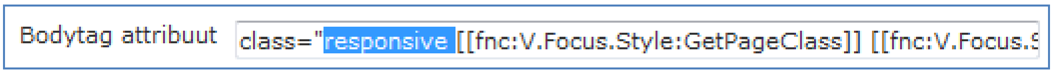
- JavaScript
- the dojo JavaScript framework
- HTML
- CSS
- XML and JSON for data exchange between server and front-end
- SWF (Flash)
- PHP
- Caché Object Script

1.2.6 Responsive design

Being responsive, Iguana will automatically optimize your existing profiles and pages to fit on each visitors screen whether that is a full display, tablet or Smartphone.

With these capabilities you don’t need to create & maintain a dedicated site profile (+pages) to manually create a mobile friendly interface.

To activate the responsive additions you must add the class ‘responsive’ to the bodytag attribute in the site profile.



Bodytag attribuut class="responsive [[fnc:V.Focus.Style:GetPageClass]] [[fnc:V.Focus.S

1.3 UUID’s

Each main Iguana component is attributed a so-called UUID to identify it. These UUID’s (‘Universally Unique Identifier’) are generated ID’s that are unique (see e.g. http://en.wikipedia.org/wiki/Universally_unique_identifier for some generic information about UUID’s).

These UUID’s are used for internal storage, but do appear in certain areas of Iguana’s CMS, and even in some areas of Iguana front-end framework (i.e. in actual sites). If such UUID’s are or need to be used to identify unique components, Short URL's may be used to create more user-friendly identifiers.

A UUID is 36 characters long and can contain letters, digits and punctuation (e.g. 204d6e2b-2b4b-44f9-a2a9-0bd3f9a6d0b7).

1.4 Language support

Iguana's language support has three aspects:

1. Front-end interface language support

2. Front-end content language support

3. CMS interface language support

Front-end interface language support

The Iguana front-end can support any language. Iguana supports multi-lingual sites, i.e. sites that have multiple language interfaces.

Front-end content language support

Iguana's support of multi-lingual sites also implies that (identical) content can be created for multiple front-end interface languages.

See the chapter *Properties and Languages* in the document "Management common workflow" for more information on how to use Iguana's CMS to create multi-lingual content and the chapter *System texts* in the document "General & Tools" for information on how to translate system texts.

CMS interface language support

The Iguana CMS supports three interface languages: English, Dutch and French.

The language that is attached to the user's profile will determine which interface language is used in the CMS.

Language fallback

Iguana makes in its front-end framework a distinction between what is called the 'active' and the 'default' language. The 'active' language is the language that the user has (implicitly or explicitly) selected as the interface language. For system texts, a fallback mechanism is implemented: if a specified text is not defined in the ‘active’ language, the application will select the specified text in the ‘default’ language.

Please note that this principle is only implemented for system texts, and not for content (e.g. not for Rich Text Items).

Language and content

For content created via the CMS, it is possible to link multiple language instances to a single content id (application id). This allows for the implementation of “fallback” logic (e.g. if language XXX of application id YYY is not available, then use the language ZZZ of that application id).

This would imply that it should be possible to create the following solution:

- a profile that has a single navigation menu
- of which the (navigation menu) elements link to a single set of pages
- show a page instance in function of the interface language
- and fall back to a default language instance if the current interface language is not defined.

Or as an example:

- in profile Kids (with user interface language = eng)
- the page with the “id” NewItems is selected
- of which the English instance is shown (if available)
- or of which the French instance is shown (if the English instance is not available, and French is the default language).

1.5 How is the user interface language handled?

Iguana’s algorithms to select a user interface language are as follows.

- The language is detected from the so-called HTTP headers (which contain information on the interface language that is derived from the user's browser settings). If this language is not included in the list of supported languages for the site, Iguana will automatically switch to the first language in the list of supported languages for the site.

- However, if the start-up URL contains a Language parameter in the URL (e.g. ?Language=eng), this language setting will be used and this will overrule the language that is detected from the HTTP headers. If this language is not included in the list of supported languages for the site, Iguana will automatically switch to the first language in the list of supported languages for the site.

- If the user switches explicitly to another language (e.g. through an explicit selection from a menu), that setting will be maintained until the user does another explicit switch to another language. The language switch will reload the current page, but in the newly selected language.

This information is related to the actual Iguana application, not to Iguana's CMS.

2 Getting started

Before you can start using Iguana and create your website, the following steps need to be taken.

- Step 1: The system needs to be installed.

- Step 2: The system needs to be configured using the Iguana CMS.

- Step 3: The site needs to be set up and content defined

2.1 Configuration and Requirements

Before you can start using Iguana or the CMS that can be used to configure the system, the server software needs to be loaded and a basic configuration will be installed.

Client-side requirements

Iguana can be used on PC's that conform to the following requirements.

- supported browsers are Microsoft Internet Explorer, Mozilla Firefox, Google Chrome and Apple's Safari

- a recent version of these browsers needs to be installed; the minimal versions are mentioned in the chapter Supported browsers in the document “Technical requirements and advice”

- a Flash plug-in is mandatory if Flash animations are included in a site’s content; the minimal required Flash version is version 10; however, if you include animations in a site’s content and there is no Flash Player installed on the user’s device, Iguana will automatically switch to a JavaScript based animation.

Server-side requirements

The system requires a data server and a web server, which can be installed on the same server machine (although this is not advised).

Iguana runs on Windows, Linux and Unix servers. Both Apache and Microsoft IIS are supported webservers.

More information

More technical information can be found in the document “Technical requirements and advice”.

2.2 Starting the Iguana CMS

The URL to start the Iguana CMS is:
http://ServerAddress/Application/**CMS.Start.cls?Language=**Language

in which

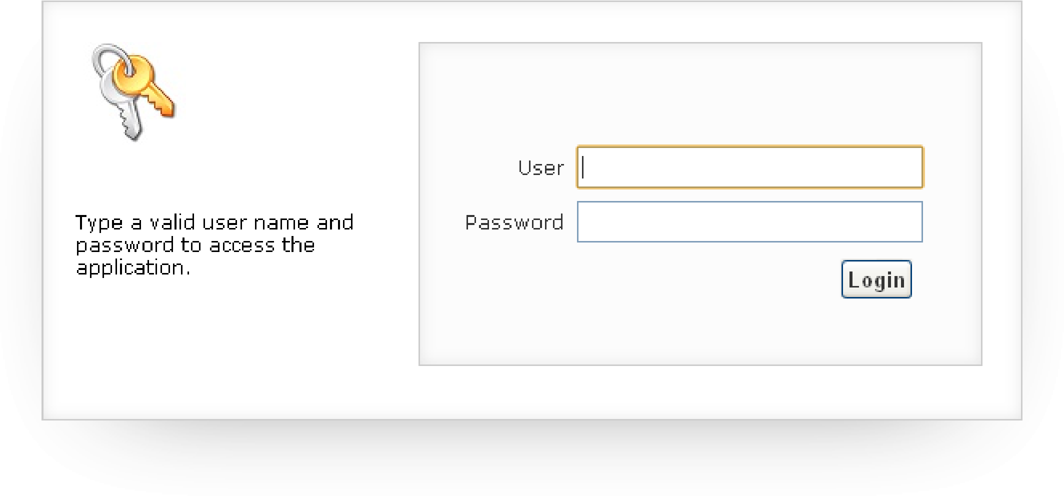
- ServerAddress = the address of your server, this can be an IP address (e.g. 10.34.322.27) or a server name (e.g. library.server) – this value is specific for each installation

- Application = the application name that is defined for the Iguana CMS in the web server configuration (e.g. IGUANA) – this value is specific for each installation

- Language = the interface language; you can select from eng (English), fre (French) and dut (Dutch); if the Language= parameter is omitted, the system will use English as the default language

3 Logon to the CMS

To log on to the Iguana CMS you have to enter valid logon credentials. To log on Iguana uses an authentication tool called Argus.

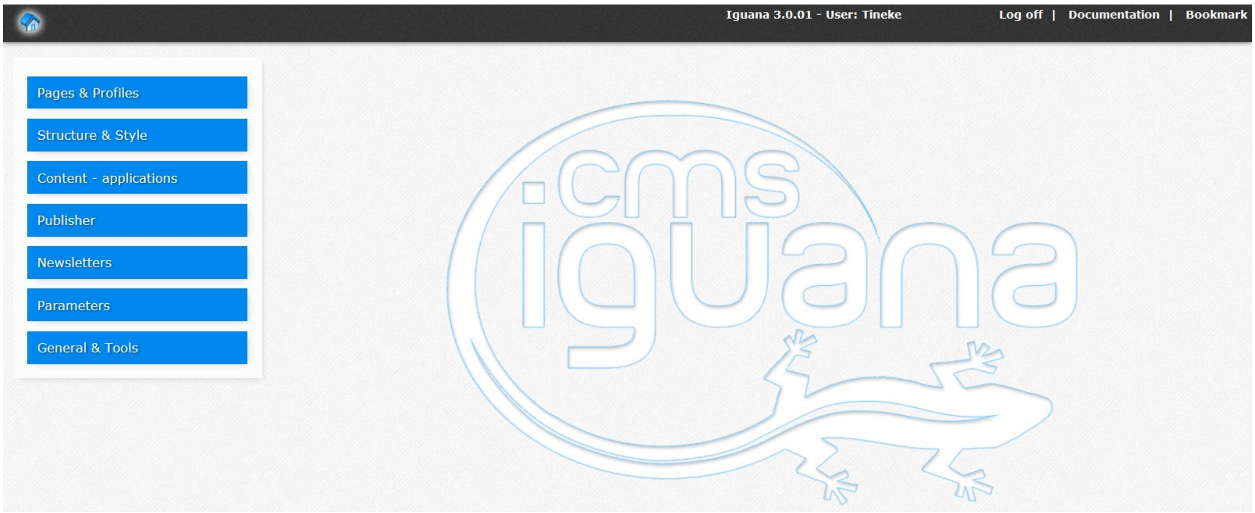


Type a valid user name and password to access the application.

The Argus documentation describes, amongst others, how you can logon, how SSO (Single Sign On) can (or cannot) operate in your environment, what you should do if logon is unsuccessful, and other topics.

4 Home page

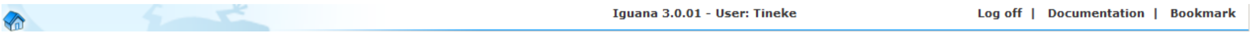
After you have gained access to the application, the system shows you the home page of the application. This is the main overview page:



4.1 Home page structure

The CMS home page has three main components.

- Top pane – The top pane gives access to the following generic options:



- Home – Return to the Iguana home page

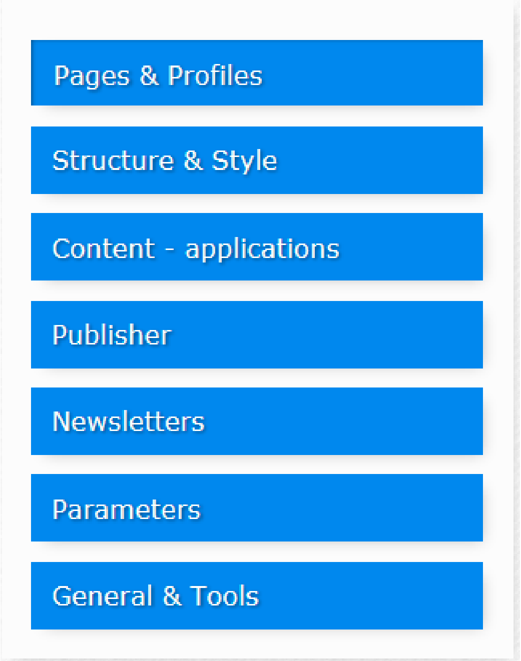
- Current user [text only]

- Log off – Allows you to log off and return to the logon page

- Documentation – Access to the Iguana online documentation

- Bookmark – Allows you to bookmark specific pages; the bookmark will be added to the “Favourites” section of the navigation menu

- Left navigation pane – The left navigation pane contains the main navigation options, grouped in categories. The full list of navigation options is included below. The navigation consists of the following sections:



- Pages & Profiles – allows you to manage site profiles, location profiles, a site’s default page structure, navigation menu’s and similar topics

- Structure & Style – allows you to manage templates

  - Content - applications – allows you to manage the “applications” that are placed on pages in a site; these applications are the placeholders of the actual content; examples of such applications are: Rich Text Items, RSS feeds, word clouds, animations, and many others

  - Publisher – allows you to define and publish content such as news items, calendar items, blog entries and similar content

  - Newsletters – allows you to configure various types of newsletters to be sent to your customers

  - Parameters – the general parameter setup

  - General & Tools – allows you to define the general parameters for a site (such as email settings and texts), to manage users and roles

- Main body pane – the main Body pane will display the actual content (as selected from the navigation pane; on the Home page the Body pane displays the main options as icons; you can click on these icons to access the option.

Please note

Only menu options for which you have a license will be visible. I.e. some sections of Iguana require an additional license on top of the standard Iguana license.
E.g. selfcheck and age groups are part of the V@school license.

4.2 Options

The options on the CMS home page are:

- Select from the navigation pane

- Log off

- Access the documentation

- Return to home page: click on the Home icon

- Bookmarking

Select an option from the navigation pane

The menu options are grouped into categories. If you click on a category header, the category will open and its content will be displayed. At the same time any other category that might be "opened" will automatically collapse.

Log off

If you click on the Log off button, the system returns to the Logon screen.

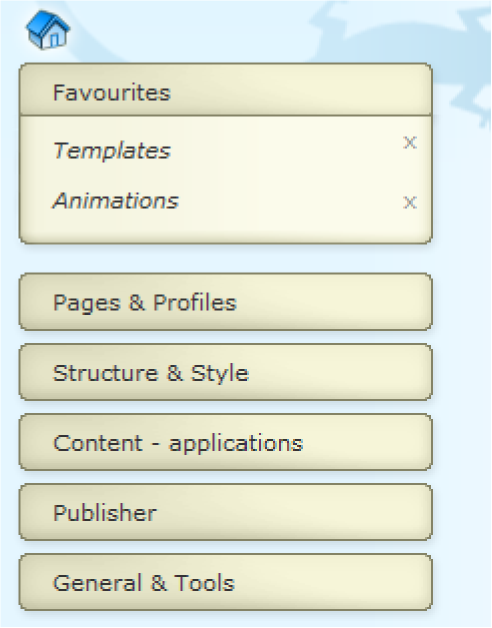
Access to documentation

If you click on the Documentation link, this will open the Iguana documentation.

Bookmarking

If you click on the Bookmark option in the header pane, the current page will be added to the list of Favourites. The list of Favourites is the first category that is displayed in the navigation pane. The Bookmark option does not ask for a confirmation. The current page will be automatically added.

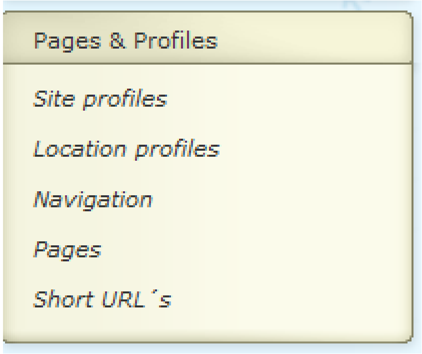
If you want to remove an option from the list of favourites, you can do so by clicking on the Delete option (the "X" icon) displayed after each favourite (in the navigation pane).



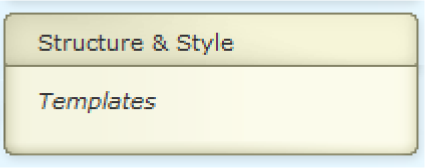
4.3 Menu options

The following options are available.

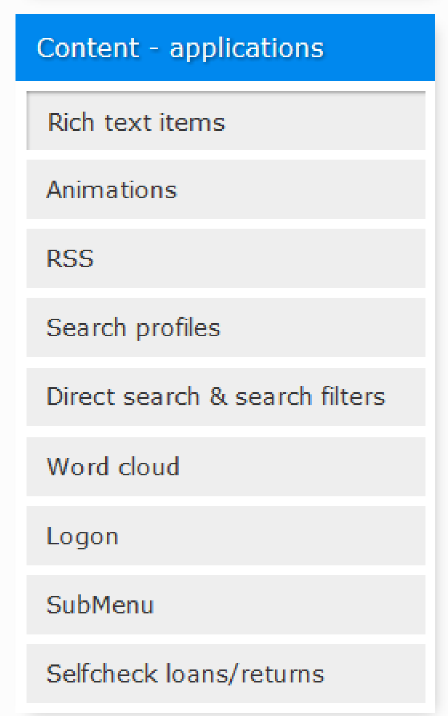
Pages & Profiles



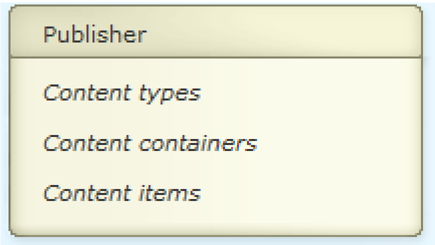
Structure & Style



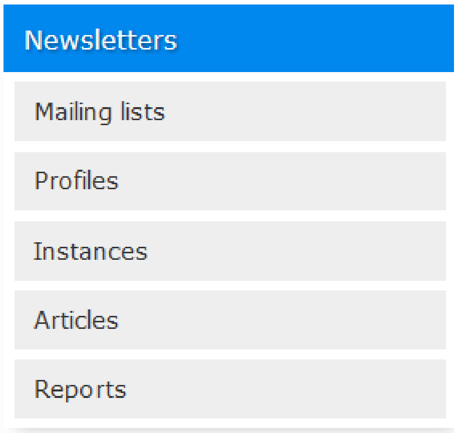
Content - applications



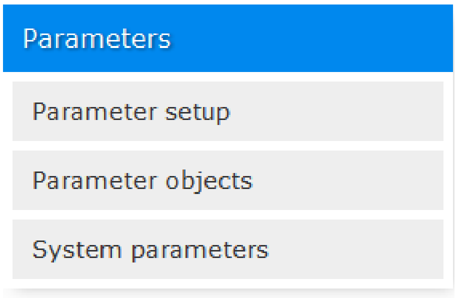
Publisher



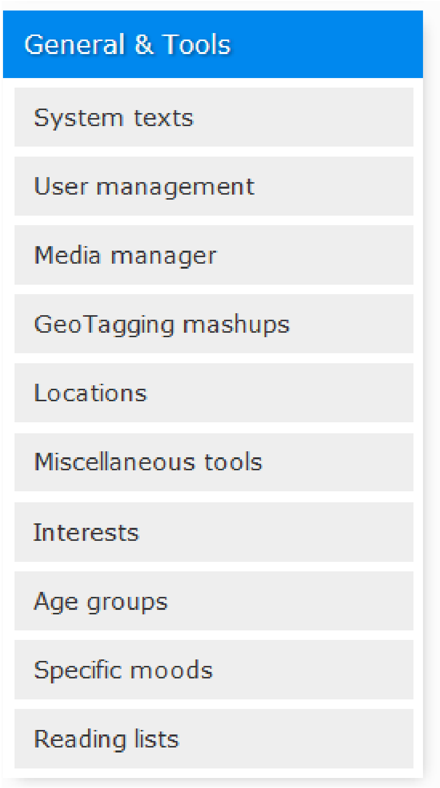
Newsletters



Parameters



General & Tools


General & Tools
System texts
User management
Media manager
GeoTagging mashups
Locations
Miscellaneous tools
Interests
Age groups
Specific moods
Reading lists

- **Document control - Change History**

Version	**Date**	**Change description**	**Author**
1.0	August 2011	Creation	
1.1	December 2011	Review	Edwin van der Geld
1.2	December 2011	Review	Debbie MacLeod
1.3	January 2012	Added information on user interface language handling	
1.4	January 2012	Added section Main technical concepts; Added header	
1.5	January 2012	Moved details on supported browsers to Technical requirements chapter Reviewed whole document	
1.6	January 2012	Minor change to section on Language detection	
2.0	May 2012	Reformat for online help doc	
3.0	August 2012	Screen shots; new options; removed incorrect reference to content on "external sites" part of 3.0 updates	
3.1	April 2014	Parameter maintenance moved to separate section; new section Newsletters part of 3.0 updates	
3.2	August 2014	New screen shot for content - applications part of 3.0 updates	
4.0	April 2016	Some textual improvements part of 4.0 updates	