

Meme Media Laboratory  
Hokkaido University

MEME MEDIA LABORATORY

Hokkaido University



WEB EXTENSION OF MEME MEDIA ARCHITECTURE

# Webble World 3.1

## User's & Developer's Guide

# Webble World 3.1 User's & Developer's Guide

---

## CONTRIBUTORS

### Webble Chief System Developer

Micke Nicander Kuwahara

### Meme Media Project Leader & Chief Architect

Yuzuru Tanaka

*This document was last updated  
January 7, 2016 12:24*

### Webble World

*[IntelligentPad system for the web]*

Copyright © 2010 Micke Nicander Kuwahara, Giannis Georgalis, Yuzuru Tanaka  
in Meme Media R&D Group of Hokkaido University

\*\*\*\*\*

Webble World is licensed under the Apache License, Version 2.0 (the "License");  
you may not use this file except in compliance with the License.  
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software  
distributed under the License is distributed on an "AS IS" BASIS,  
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
express or implied.

See the License for the specific language governing permissions and  
limitations under the License.

The use of the word "Webble" or "webble" for the loadable meme media objects  
are limited only to objects that actually loads in this original Webble World  
Platform. Modifications of the meme media object code which leads to breaking of  
compatibility with the original Webble World platform, may no longer be referred  
to as a "Webble" or "webble".

For any further information, visit the project Wiki at

<https://github.com/truemrwalker/wblwrl3/wiki>.

\*\*\*\*\*

Meme Media R&D Group  
Kita 13, Nishi 8 Kita-ku, Sapporo  
060-8628 Japan  
Phone +81-(0)11-706-7250  
[mkuwahara@meme.hokudai.ac.jp](mailto:mkuwahara@meme.hokudai.ac.jp)

# Table of Contents

Table of Contents .....	2
Preface.....	7
Introduction .....	8
What is a Webble? .....	8
But what are Webbles, really? .....	9
Who makes primitive Webbles? .....	9
What do you do with all these primitive Webbles? .....	9
User's Manual .....	11
Quick Start .....	11
Webble World Activities and "Webblish" terms .....	20
Assign/Revoke Parent .....	20
Browse .....	21
Bundle .....	21
Cancel .....	21
Child .....	21
Connect Slots.....	21
Connection View .....	21
Custom Slots.....	21
Delete.....	21
Developers Pack.....	22
Duplicate .....	22
Execution State / Execution Level .....	22
Interaction objects (balls).....	22
Load .....	22
Login & Register .....	22
Main Menu .....	23
Merged Slot.....	23
Parent .....	23
Properties.....	23
Protection .....	23
Publish a Webble .....	23
Redo .....	24
Shared Model Duplicate .....	24
Slots .....	24
Super parent .....	24
Undo .....	25

Upload and Edit Template .....	25
Webble .....	25
Webble Id.....	25
Webble Menu .....	25
Webble Prototype .....	26
Webble Template.....	26
Webble World Work Surface .....	26
Workspace .....	26
Webble World System Menu.....	26
Workspace ->New .....	26
Workspace ->Open .....	26
Workspace ->Save .....	27
Workspace ->Save As .....	27
Workspace ->Share .....	27
Workspace ->Delete .....	28
Workspace ->Print .....	28
Webbles->Browser .....	28
Webbles ->Load.....	29
Webbles ->Recent .....	30
Webbles ->Publish.....	30
Webbles ->Template Editor .....	30
Webbles ->Sandbox Webbles (listed by their names).....	31
Edit ->Undo.....	31
Edit ->Redo .....	32
Edit ->Select All .....	32
Edit ->Deselect All .....	32
Edit ->Duplicate .....	32
Edit -> Shared Model Duplicate .....	32
Edit ->Bundle .....	33
Edit ->Delete .....	33
Edit ->Multi-Webble Properties.....	33
Edit ->Platform Properties.....	34
View->Toggle Connection View.....	35
View ->Active Workspace info .....	35
View ->Toggle Full Screen.....	36
Help->Documentation .....	36
Help->FAQ .....	36
Help->Open Chat.....	36
Help ->Support.....	37

Help ->Developer Community.....	37
Help ->Download Developers Package.....	37
Help ->Webble Platform on GitHub .....	37
Help ->Report Bug .....	38
Help ->About.....	38
Sign Up .....	38
Login .....	39
Username-> Profile Settings->Account Information .....	40
Username-> Profile Settings ->Notifications.....	40
Username-> Profile Settings ->Trust .....	40
Username-> Profile Settings ->Authentication .....	41
Username-> Groups .....	42
Username->Logout .....	43
Language .....	43
Default Webble Interaction Balls .....	43
Open Menu .....	43
Rotate .....	43
Resize .....	44
Assign/Revoke Parent .....	44
Webble Default Popup menu .....	44
Publish .....	44
Duplicate .....	44
Delete.....	44
Assign Parent.....	44
Revoke Parent .....	45
Connect Slots.....	45
Properties.....	45
Bundle .....	47
Unbundle.....	47
Shared Model Duplicate .....	47
Bring To Front .....	47
Set Protection .....	47
Add Custom Slot.....	49
Custom Menu Items.....	50
Custom Interaction Objects .....	50
About.....	51
Short-Cut Keys.....	52
Esc.....	52
Enter .....	52

ALT + F1 .....	52
ALT + F2 .....	52
ALT + F3 .....	52
ALT + F5 .....	52
ALT + F6 .....	52
ALT + F8 .....	53
ALT + F9 .....	53
ALT + F10 .....	53
ALT + SHIFT + Page Down .....	53
ALT + SHIFT + END .....	53
ALT + Arrow keys .....	53
ALT + N .....	53
ALT + O .....	53
ALT + S .....	53
ALT + J .....	53
ALT + X .....	53
ALT + SHIFT + P .....	53
ALT + B .....	53
ALT + L .....	53
ALT + R .....	53
ALT + P .....	53
ALT + U .....	54
ALT+ Z or CTRL + Z .....	54
ALT+ Y or CTRL + Y .....	54
ALT + A .....	54
ALT + SHIFT + A .....	54
ALT + D .....	54
ALT + SHIFT + D .....	54
ALT + Delete or CTRL + Delete .....	54
ALT + E .....	54
ALT + SHIFT + E .....	54
ALT + Numpad 9 .....	54
ALT + I .....	54
ALT + F .....	54
ALT + M .....	54
ALT + SHIFT + F .....	54
ALT + C .....	54
ALT + SHIFT + C .....	54
ALT + H .....	55

ALT + G.....	55
ALT + SHIFT + G .....	55
ALT + Numpad 7.....	55
ALT + Numpad 5.....	55
ALT + Mouse / ALT + SHIFT + Mouse .....	55
Mouse Interactions.....	55
Touch Screen Gestures .....	55
Swipe Down [on the work surface] .....	56
Swipe Up [on the work surface] .....	56
The Webbles .....	56
Summary.....	56
Developer's Manual .....	57
Webble Architecture.....	58
Webble Parts .....	58
Webble Instantiation .....	58
Developing a Webble .....	59
Creating a Webble .....	59
Tips & Tricks & Basic Stuff.....	60
Custom Forms .....	62
Custom Messages .....	62
Cool custom JavaScript or AngularJS .....	62
Web Services, data bases and APIs' .....	62
Summary.....	63
Bibliography .....	66

# Preface

*A few words from the creators.*

Here we are pleased to release our fifth generation IntelligentPad system, 'Webble World 3.1,' and this 'Webble World User's & Developer's Manual', which is focused both on trial use for evaluating this new knowledge media architecture that is fully compatible with the Web and for the extensive use in developing advanced Web applications and sharing such applications as shared components with other people for their further application development efforts.

Webble World 3.1 is a combined extension of the two- and three dimensional meme media architecture of IntelligentPad and IntelligentBox. Meme Media Laboratory of Hokkaido University has been conducting R&D efforts to develop a new knowledge media architecture that allows us not only to publish compound documents with or without embedded services and tools into a world-wide shared repository like the Web, but also to reedit any compound documents available from such a shared repository, through direct manipulations, and through composition of new compound documents for local reuse and for publication to the shared repository. The reediting of compound documents denotes the extraction and recombination of their visual portions to compose a new compound document with a new layout and new interoperability of the extracted components.

The compound documents accumulated in such a world-wide repository may work as genes to evolve their variety. They can be easily copied, which corresponds to the self-replication of genes. Their reediting by people corresponds to the recombination of genes. Some people may replace a component of a compound document with another to find out an unexpected functional effect. This may correspond to a mutation of a gene. Finally, they are subject of people's evaluation. Some of them are frequently copied and reused, while others are gradually forgotten. This corresponds to the natural selection of genes.

Based on this similarity between such compound documents in a shared repository and genes, we named our knowledge media architecture 'meme media architecture.' This word 'meme' was coined by Richard Dawkins in 1976 [Dawkins 1976] to denote a cultural counterpart of a gene for focusing on the similarity between biological and cultural evolution.

Meme Media Laboratory has developed two different types of meme media architectures, namely, IntelligentPad architecture for two dimensional representation of contents, and IntelligentBox architecture for three dimensional representation of contents.

The first generation of IntelligentPad architecture used Smalltalk 80, and was released in 1989 from our group. Its second generation used C++, and released in 1993 from IntelligentPad Consortium member companies, Fujitsu and Hitachi Software in



cooperation with our laboratory. The third generation that was released in 2000 by a US based venture company, K-Plex Inc. in cooperation with us used C# and XML description of components, and was developed as a Web-top system and the forth generation, and just previous to the current, was fully available via any main web browser using Microsoft Silverlight technology.

For more details of meme media architectures, technologies, and applications, we recommend the following book (Tanaka 2003) as your reference:

*Y. Tanaka. Meme Media and Meme Market Architectures: Knowledge Media for Editing, Distributing, and Managing Intellectual Resources. IEEE Press & Wiley-Interscience, NJ, 2003.*

Webble World is designed and developed based on this long history of our R&D experience. It is a Web-top system, requiring no need to install any client kernel in each PC, fully implementing the philosophy and architecture of meme media, fully upper compatible with the current-state-of-the-art of Web technologies, and enabling us to deal not only with any visual functional component with a canvas, but also any graphical functional object without a canvas. The last feature allows us to edit graphics as a compound object composed with graphical and functional primitive objects.

We will very much appreciate if you will participate in our activity to collaboratively improve and enrich this Webble World 3.1, and to explore new areas and new ways of its applications, by exchanging and sharing our ideas and achievements. We have recently just released the latest fresh update of Webble World 3.1 over the Web, the primary meeting plaza in the Webble Universe. We welcome you to visit this page, and to enjoy playing in this new universe of experience.

March 14th, 2014

*Meme Media Project Leader and Chief Architect*  
*Director of Meme Media Laboratory*

**Prof. Yuzuru Tanaka**

*Webble Chief System Developer*

**Micke Nicander Kuwahara**

## Introduction

*Get to know what a Webble is and what it can do for you as a knowledge seeking, culture sharing, thought mongering Internet user.*

Internet is the place where everything merges together, every thought, every dream every face. It's where you find our history our present and our future. It's the place where every voice may be heard and every idea may find a follower. It is there that so much begins and nothing ever ends. Internet is the steppingstone and platform for freedom in any form and shape. It's there where we share and communicate the knowledge of our universe. It is the universe within the universe.

But as our understanding of culture and knowledge sharing gains, the need for new, more open and more powerful tools emerges, tools that fulfill our need to build and communicate, the need to be inspired and further inspire others, the need to develop and change, edit and contribute. A tool that helps each and every one of us realize that the only thing that is constant is our desire to evolve.

A tool like that needs to allow all types of users to participate, from any background and with any sets of skills. It needs to be open and free so it may evolve and grow alongside the rich source of knowledge it is carrying. It should be easy to access from anywhere in the world and it should adapt to our every need with only one internal goal; to share the collected minds of us all.

Nothing is ever finished, and what will be shown to you within this document is not any exception, but it has definitely begun, right here and right now with The World of Webbles.

### What is a Webble?

Webble is the acronym of the word WEB-PEBBLE, where the WEB part is of course just an indication where these Webbles operate and PEBBLE stands for Pad Enhanced Building Block Lifelike Entity. So what does Pad Enhanced mean? It means that the Webble Architecture is based on the IntelligentPad Architecture designed at the Meme Media laboratory at Hokkaido University by Professor Yuzuru Tanaka which regards knowledge media for editing, distributing and managing intellectual resources. The scope for this document is not to explain what an IntelligentPad is, so for deeper understanding

on that subject I must point you in the direction of Professor Tanaka's book (Tanaka 2003) found in the bibliography at the end.

So basically a Webble is an IntelligentPad with some additional functionality and enhanced features, within the framework of IntelligentPad architectural design, which exists and operates in an Internet browser over the Web.

Webbles also have the potential to evolve naturally into an IntelligentBox (3D version of an IntelligentPad) without any major change in the current working Webble World 3.1 System. That possibility comes from the fact that the visualization and user interaction of a Webble is built with the latest web technology of HTML5 and fully uses the power of that technology as it becomes even more powerful with every iteration.

### **But what are Webbles, really?**

They are primitive standalone objects of versatile shape and usage capabilities that can be loaded inside a Webble World 3.1 System running on an Internet server via a browser and then manipulated by the user to look and behave in any way the user wants. These Webbles can then be combined into complex compound Webbles, where each Webble communicates with the others via slot connections and can then create more advanced behaviors. Many Compound Webbles may also be combined into applications. These creations done by single users can be saved and later loaded by other web users who can use them or edit them and resave them into new creations. Only imagination is the limit of what you may create and share with Webbles and how they can be used.

### **Who makes primitive Webbles?**

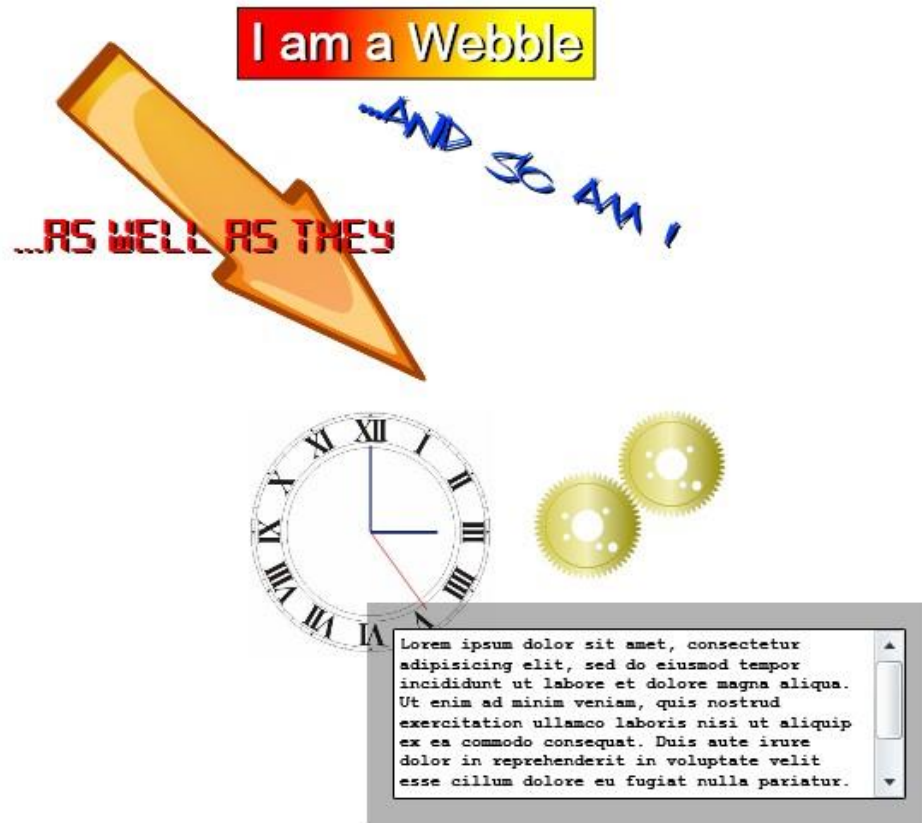
Webbles templates (aka prototypes) are created by anyone with basic web developing skills, using HTML, JavaScript and CSS, preferably with the integrated JavaScript-based system framework of AngularJS, using the provided Webble Element template skeleton. A Webble may be easily manufactured within an hour and contain only some basic behaviors and usage points or it may be a venture for a group of developers to develop an extremely powerful Webble with many versatile tricks up its sleeve. It is all up to the developer and what he or she wants the Webble to be able to do. Finished Webble-templates are then published to the public for download, and as the number of Webble developers grows the number of available Webbles grows even more. The future is wide open for the Webble developers to create new Webbles prototypes and easily add them for use in the online Webble Repositories.

### **What do you do with all these primitive Webbles?**

As an Internet user you surf to a Webble World page and there you find this vast sea of Webbles just waiting for you to load into the browser and then use them. In a not too long future, you will probably even be able to google for Webbles as any other online information. There might be plain text Webbles, image Webbles or button Webbles or more complex ones like database communication Webbles, Web clip Webbles and 2DGameSprite Webbles. As the user grabs hold of them into his or her browser he can

start filling them with his imagination. They can be combined into more complex compound Webbles that solve some specific tasks and when they are done they may be saved and shared with the rest of the world. Also compound Webbles may be further edited and combined into some real serious web applications. So no matter if you just want to share a text, image or movie or if you want to create a new hit game, virtual lab application or the best online office tool ever made, Webbles is your key. And as the number of available Webble prototypes increases the stronger and more powerful the Webble World Designers can become and the tools and applications created for the world Internet users will boom to heights unimaginable.

So let's Webble away...



## User's Manual

*The Webble World is now running in your browser, but what is that boring empty background color, and what are you supposed to do with that menu thing on the top? Where are the all the Webbles you were promised? Should you demand a refund? Of course not – it is freeware, so that will not do you any good. It is better to learn how to use the Webble World System so that you can blow our minds away with your creations.*

In this chapter We will try to explain everything that you need to know about being an Internet user of The Webble World system, how to load Webbles, how to combine them, how to manipulate them and how to save your creations. In order to learn this you may also need to learn a few new terms and expressions, and in the worst case some new ways of thinking, but do not worry all will be explained in detail and you can also find a “Webblish” wordlist in this chapter that might be helpful. So let’s “Webble” away...

### Quick Start

For all of you out there who just are not that into manuals and want to get right into it as soon as possible, we start off with a simple step by step Quick Start description that lets you get the simple basics of Webble World and how to navigate there.

1. To load our first Webble we click ‘Webbles-> Browser’ in the top menu.

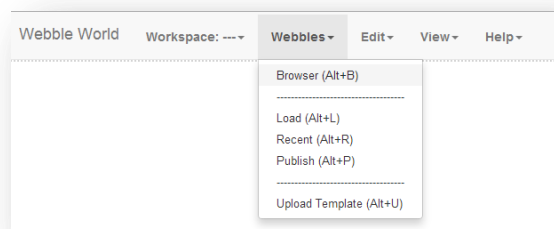


Figure 1: Open the Webble Browser, which contacts the Webble Web Service

2. Write 'fundamental' in the search box and click 'Search'

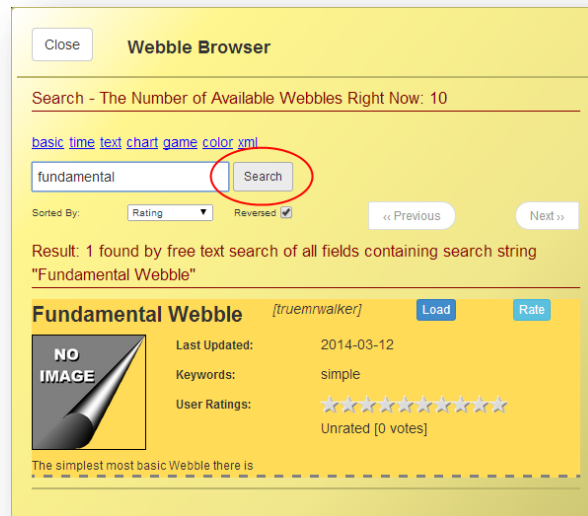


Figure 2: The Webble Browser in action

3. Select the 'Fundamental Webble' and either click 'Load', double-click the Webble area or drag the Webble image outside the form to load it, and then 'Close' to close the Webble Browser form.
4. The little gray square in the left top corner of the Webble World desktop is our Fundamental Webble. Not much to look at, but perfect to start with.

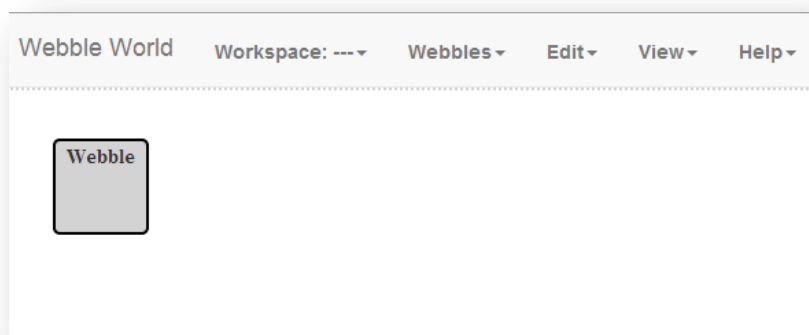


Figure 3: We have loaded our first Webble

5. If you double-click on the gray Webble it will become selected and show you its name. Double-click again to make it non-selected.

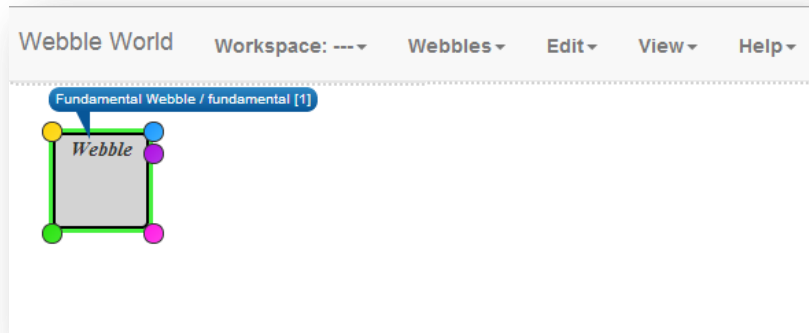


Figure 4: The info-bubble disappears after a few seconds.

6. Selected or not, if you click the gray Webble and drag the mouse you can move the Webble to anywhere you want inside the Webble World Desktop.
7. Let's change the look of our little gray friend by opening the Webble Properties form. We do that by first opening the Webble Pop-up menu which we get if we either click on the yellow Interaction Ball in the top left corner of the Webble or right click anywhere on the Webble shape. The Interaction ball grows slightly when hovered over and it can then be clicked. In the Menu, select Properties with the mouse pointer and click again.

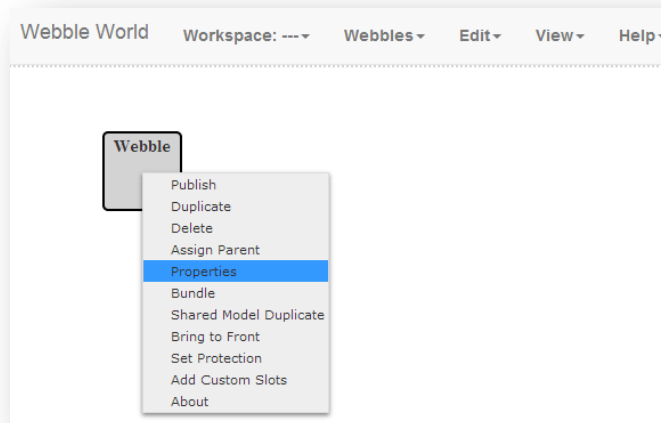


Figure 5: Yes, there is a lot you can do, but stay with properties for now.

8. The property will then display all the properties this particular Webble possesses. You can scroll the form window to see all properties, which are grouped into logical categories. Webble properties are almost always slots, meaning that they may also be connected to and can communicate with other Webbles' slots.

Close Submit **Properties**

Internal Slots: Internal

Display Name: Fundamental Webble ?

Slots: fundamental

Message: Webble ?

Slots: CSS Related (fundamental)

Width: 70px ?

Height: 70px ?

Background Color: #D3D3D3 ?

Border: 2px solid rgb(0, 0, 0) ?

Border Radius: 5px ?

Font Size: 15px ?

Foreground Color: #333333 ?

Font Family: times new roman ?

Slots: CSS Related (root)

Layer Order Index: 10 ?

Horizontal Position: 82.00px ?

Vertical Position: 53.00px ?

Figure 6: You may change any property you want, and all changes take effect when the form is submitted

9. Change Height to 200, Background Color to Red (by clicking the color box and select a color or by typing the color hex value for red (`#ff0000`)), Border Color to Blue in the border box by changing the `rgb(0, 0, 0)` part to either `#0000ff` or `rgb(0, 0, 255)`. You may also change the text in the message slot if you want, it is all up to you.
10. Click 'Submit' and your Webble will change according to your request. Close the property form by clicking "Close".

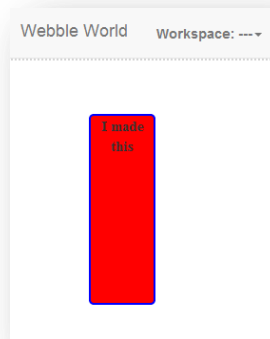


Figure 7: Looks a bit nicer, doesn't it?



11. If the Webble is selected (if not, select it again), click 'Edit' in the top main menu and then click 'Duplicate'.

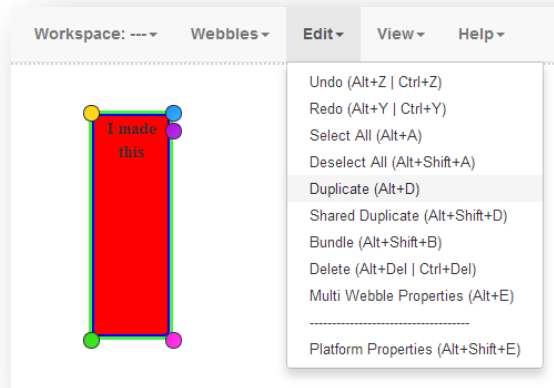


Figure 8: Duplication may also be done via the Webble's context menu, but for now we go via the main menu.

12. Repeat the duplication one more times so there will be three identical Webbles in total and then move them slightly so that you may see them all at once.

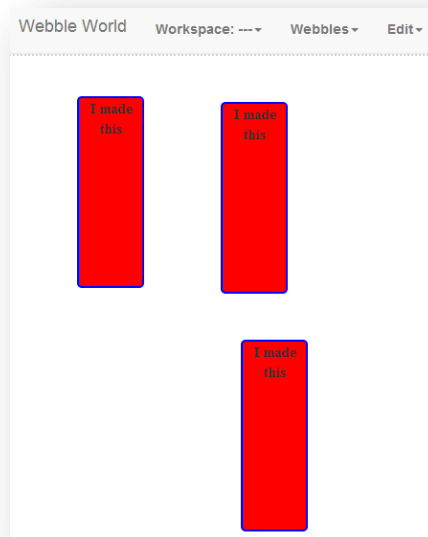


Figure 9: Three independent Webbles ready for duty.

13. If you place the Webbles as above and then select the top right one, you may create a child - parent relationship with the one on its left by clicking the green ball in the bottom left corner of the Webble and then select a possible target

(edges glowing blue) by clicking on it to create that relationship. This is briefly indicated by the parent glowing deep pink and the child glow in light pink.

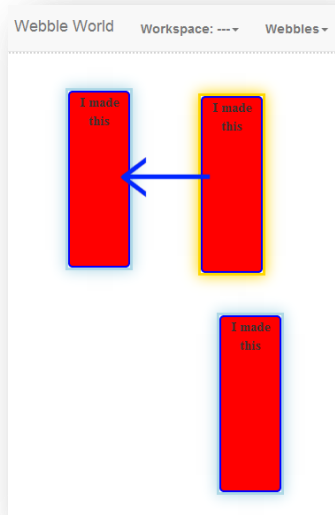


Figure 10: Making a child - parent connection.

14. Create one more child - parent relationships as indicated in the figure below. When you are done try moving the Webbles slightly to see how they are attached to each other.

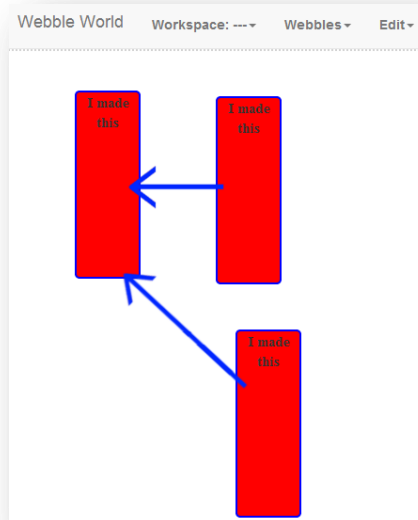


Figure 11: The arrows indicate child-to-parent connections. A child may only have one parent, but a parent may have many children.

15. Select the top right Webble and open its menu and there click 'Connect Slots'.

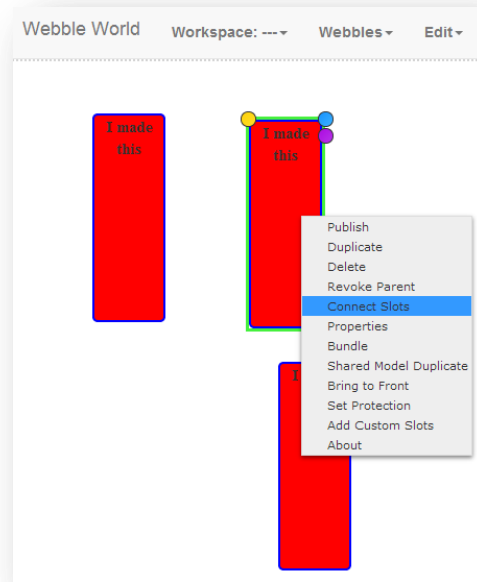


Figure 12: A Webble needs to be a child in order to have the 'Connect Slots' menu item available.

16. In the Slot Connection form you may pick which child slot should be connected to which parent slot and also whether the communication should send and/or receive. By default the slot connection is already set up for the message slot (text), but we will change that. Select the Width slot for the child and the Height slot for the parent and leave the direction for both send and receive. When finished you click 'Submit' to activate the slot connection.

 A screenshot of the 'Slot Connection' dialog box. It has a yellow background and a title bar. At the top left are 'Cancel' and 'Submit' buttons. The title 'Slot Connection' is centered. Below the title, there are three rows of controls:
 

- Child Slot:** A dropdown menu showing 'Width [square:width]' with a question mark icon to its right.
- Parent Slot:** A dropdown menu showing 'Height [square:height]' with a question mark icon to its right.
- Connection Directions:** Two checkboxes, 'Send' and 'Receive', both of which are checked. A question mark icon is to the right of these checkboxes.

Figure 13: The child and the parent slot do not need to be the same but they could be that too.

17. We have now created a slot connection between the top right Webble and its parent to the left, telling them to share, both directions, child width and parent

height values. This is immediately seen and understood by the fact that the parent Webble suddenly became square again, indicating that the height of the parent corresponds to the width of the child. You can also test this by selecting either one of the two and then clicking and dragging the pink Interaction object in the bottom right corner which resize the selected Webble, transforming that Webble's height and width. And that in turn should cause the other one to resize as well, at least the part connected. If you resize any Webble without that slot connection, of course only the Webble itself resize.

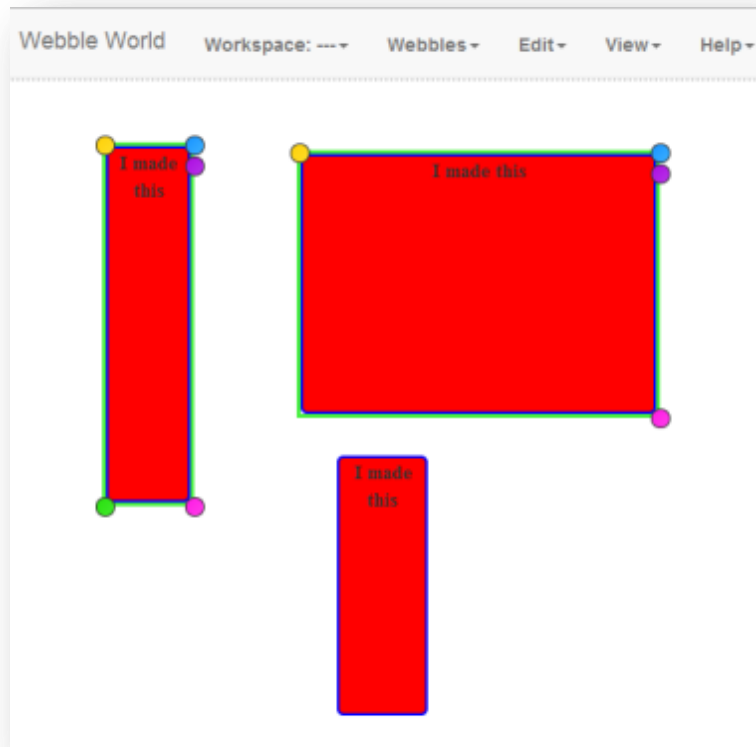


Figure 14: Resize width and height of child and parent via slot connection.

18. Slot connection is very powerful and depending on slots and Webble types and parent-child connection setup one can create quite interesting behavior. If you for example connect slots between the lower Webble using 'Message' for parent and 'Height' for child (dual communication) and then change the slot communication for the top right Webble to instead be 'Message' in the parent (keep 'Width' for child, then when you resize using the either the upper or lower right Webble the Webble below or above will follow accordingly while the parent will just display the value they share because of the chain of communication going on.

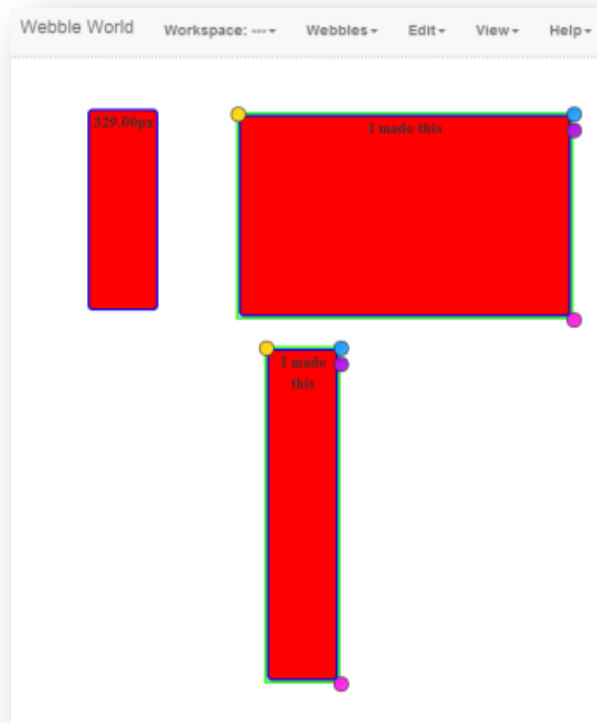


Figure 15: Slot communication is a core part of the architecture.

19. You can also change the lower right Webble connection to use 'Font Size' as the child slot for a completely different behavior.

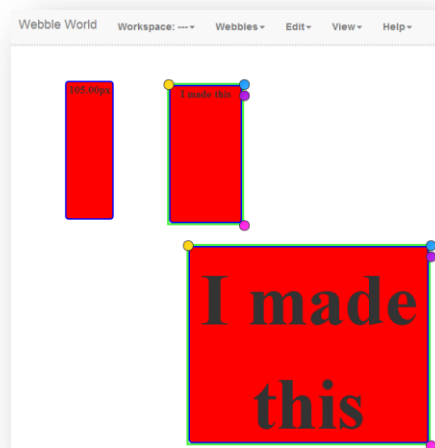


Figure 16: As you see there are basically just your imagination and the power of the slots that limits you.

20. This should be enough to grasp the fundamental parts. Now, one may experiment by loading other Webbles from the Webble browser and try them out and investigate how they work. There are not an unlimited number of Webbles available, but many enough to learn the basics of Webble World.

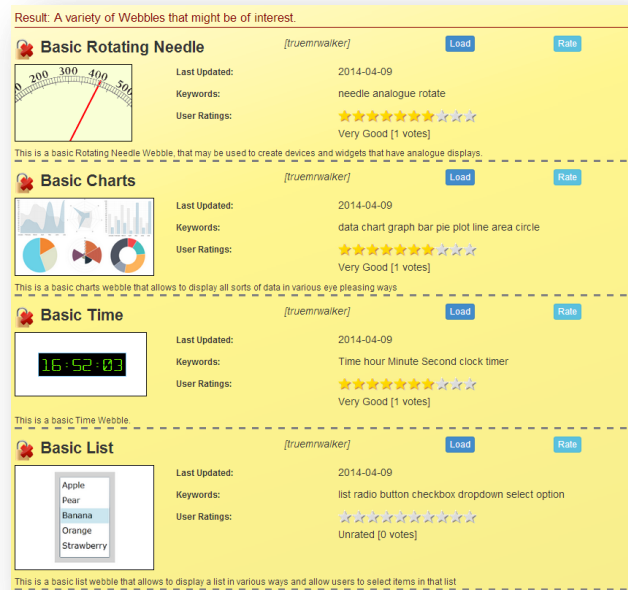


Figure 17: If you look for a Webble that is not there, maybe you can build and publish it yourself.

21. That's it, and that's that, just me talking Webbles. For detailed Webble knowledge and how to save your creations please continue reading.

## Webble World Activities and "Webblish" terms

This section explains in detail and alphabetical order all the basic different activities one might do inside the Webble World and why.

### Assign/Revoke Parent

Webbles connect with other Webbles by becoming children or parents of each other. A Webble may have many children, each of which may have many children, and so on, creating grandchildren and grandparents or just plain families and descendants. However, a child may only have one parent. All slot communication follows this chain of command. One always assigns a parent by first selecting the Webble that will become child, and then the parent. The assigning action is visualized in the Webble World System by a golden glowing border of the child to be, and a blue glowing border of possible parents (Figure 10), which turn different shades of pink when relations is made. Children cannot become parents to their own parents in some sort of circular incestuous relationship. A Child can at any time revoke a parent and pick a new parent, but any communication with the original parent will then of course be lost.

**Browse**

Browsing basically means seeing a list of available Webbles stored on the server, inside the Webble Browser Form (Figure 17). The Webble browser lists Webbles by name, popularity and developer and it present each Webble in detail with description, user rating and image. You can then select one Webble at a time to load it into your Webble World. If the Webble were created by you, you may also select it for deletion. Delete should be used with care since they will be gone forever.

**Bundle**

A Bundle is a group of Webbles bundled together as one, with its own customized input and output configuration and where the containing inner Webbles are not accessible on their own. This is a great way to package a complex Webble before publish in order to make it easier to use and manipulate. Any Webble bundle can at any time be unbundled in order to get inside to the smaller parts.

**Cancel**

Any activity may be canceled by either pressing a visible cancel button or by pressing the Esc key on the keyboard. The latter will close all open forms, cancel any parent-assigning operation, and deselect all Webbles.

**Child**

A Child is a Webble that has a parent.

**Connect Slots**

A slot connection is made via the Slot Connection form (Figure 13) where one chooses a child slot and a parent slot that will exchange values. The direction of the communication may also be chosen, either send, receive or both. Only one slot can be connected with one parent. But one parent may have many children with different slot connections. There are merged slots though that is one and many at the same time. Slot Connection is always controlled and selected via the child.

**Connection View**

The connection view is a way for a Webble developer to get an overview of what Webble is connected with which and what Slots they communicate. This view can be toggled on and off via the Webble main menu.

**Custom Slots**

A custom slot is a slot created by a user during runtime within the Webble World platform (and not within any JavaScript code). These slots can be either plain common values as text or numbers, or they can be attached and related to visual style sheet values or they can be a package of several slots, a so called merged slot. Custom slots are also stored and saved when a user publish a Webble or save a Workspace.

**Delete**

When deleting Webbles, all selected Webbles will be deleted (along with their children, which are automatically selected too). Any connections between the remaining Webbles and the ones that were deleted will be lost as well. In The Webble World System one may also have the opportunity to delete Webbles or applications from the server, inside the

Webble Browser form, but that is a permanent action so it should not be done unless the user really knows what he or she is doing.

**Developers Pack**

This is a zip file available for download from the Webble World platform server which contains everything a future Webble developer needs in order to begin making his or her own Webbles. The pack includes documentations and reference code for HTML, JavaScript and CSS using AngularJS framework.

**Duplicate**

Duplicating means exactly what it sounds like. It will create an identical copy of each Webble that is selected. Everything gets copied from the original Webble with one exception; the parent. Slot values, Slot Connections, Children are all copied – but not the parent. A parent gets its children duplicated but a child never gets its parent duplicated. So when duplicating a Webble that depends on a parent via some sort of slot communication, the duplicate will lose that connection.

**Execution State / Execution Level**

Execution state is used to define the current user state the system is running in, and Webbles may be designed so that they behave differently depending on current state. There are six available levels, which range from ‘Developer’ mode to ‘Low Clearance User’ mode. These levels are offered to let Webble designers control user access rights of the Webbles. The default state for the system is ‘Super High Clearance User’.

**Interaction objects (balls)**

Small ball-like objects attached to the bounding frame of a Webble, visible when selected, that can be used to manipulate a Webble in some way: rotating, rescaling, assign parents and many more customizable things.

**Load**

Loading a Webble means that the user has selected a Webble either from the list of Webbles presented in the Webble Browser form (Figure 17) or via the local file system or a direct web URL and then clicked ‘Submit’. The Webble World System will then download one JSON description file followed by one view.html file an most likely but not necessary one styles.css and one controllers.js, open them all and combine them so that a Webble of the chosen kind may be created and inserted into the system desktop. A Webble may be a simple prototype Webble represented by only one set of html, css and JavaScript files or it may be a combination of many connected Webbles. The loading time is dependent on bandwidth and package size. A Webble is always loaded in the background behind the Webble Browser form so that a user does not need to reopen the form for every Webble he or she wants to load. One can load any Webble multiple times on the same desktop.

**Login & Register**

Registering a user account for Webble World will allow the user to fully participate in the Webble World experience. If users register for an account they will be able to publish and save their own Webble creations. It is completely free and is only for server security and Webble protection reasons. Webble accounts can be created using existing social network



accounts such as Gmail, Facebook and Twitter. Connecting these accounts to your Webble account will in the future allow you to post your creations and work to those sites too.

**Main Menu**

The main menu, also known as the Webble World menu, is the top menu on the Webble World web site.

**Merged Slot**

A merged slot is a slot containing two or more other (non-merged) slots wrapped up together. Since Webbles only allow one slot to be communicated for each parent, an artificial rule created in order to simplify the process of Webble structure, the concept of a merged slot solves all problems which that rule may create.

**Parent**

A Parent is a Webble that has children.

**Properties**

Basically properties are the same as slots with one exception; the customized name of the Webble. The properties or slots may be found and edited in the Webble's personal property form (Figure 6) or it can be edited in real-time in the multi Webble property manipulator form (Figure 29). Some slots may not show up in the property form because they are internal, but the user may find them in the slot connection form. Other properties may be shown in the property form but not be editable.

**Protection**

Protection can be set on a specific Webble in order to disable features and behaviors which a Webble normally can do in order to customize it even further. For example the user can disable a Webbles ability to move, resize, and assign parents and much more. This is managed in the Webble Protection form (Figure 46).

**Publish a Webble**

Publishing a Webble mainly means making it available and downloadable in the online repository. Though one can also publish a Webble only as a local file too. Technically it means creating a JSON-file of the selected Webble and giving it descriptive data in order to be found and used. This is the way Webble designers, working inside the Webble World platform, will add their contributions to the Webble World Community. You can only publish Webbles online if you are logged in, meaning you have to be a registered user (of course free). When you publish something using the same name as something you already published a new revision will be created. You can edit information and either use an auto-generated image or add your own which will be shown in the Webble Browser. In the 'Add Publishing Group' you assign which of the groups you are a member of that will take responsibility for this creation. No groups selected the responsibility is all yours.

Figure 18: Publishing a Webble can only be done by registered users.

### Redo

The concept of redo is same as in most other software. It will redo the latest Webble manipulation that was just undone.

### Shared Model Duplicate

It is similar to ordinary duplication in most ways but with one very important difference: the internal slots are not duplicated but are instead shared, meaning that the two seemingly separate individual Webbles shares the same model, and so whatever happens with one Webble's slots happens also to the other. This is a very useful type of Webble connection when creating complex systems. Two slot values are not shared by default because it would make little sense, and that is the x- and y-position of the Webble on the screen.

### Slots

Slots are attributes or properties that describe and control a Webble internally but may also be attached to other Webbles' slots (therefore the name slot; imagine a connection cable from one socket to another in an appliance of some sort, sending some sort of data). One may alter by hand the values of these properties in for example the Property form (Figure 6) in order to change the Webble but they may also be changed via slot connections. Slots are more or less the engine and driving force of a Webble. A Webble's slots are a combination of several types of slots: internal slots and style sheet slots, where the latter basically control the appearance of a Webble. The user may also add custom slots to a Webble via the Webble menu.

### Super parent

A Super parent is the topmost Webble in a parent-child relationship chain. A Super parent does not have any parent of its own, but it has many descendants (grandchildren).

### Undo

The concept of undo is same as in most other software. It will undo the latest Webble manipulation that was just performed.

### Upload and Edit Template

Upload a template means getting the code part of a newly developed Webble online for testing and further development. An uploaded template only exists within the uploading user's realm and is not yet published or visible to the outside world. It exists within the framework of Webble World though so that the developing user can combine with online published Webbles.

### Webble

A Webble is an independent IntelligentPad entity that may be loaded inside a Webble World system and modified to do tasks of some kind. (see this Guide's Introduction, Chapter 1 for more information).

### Webble Id

All Webbles have several ways to be identified, but the most unique one is the Webble ID. The ID is used mainly for internal use and will change if the Webble is deleted and reloaded, but are sometimes used as a supportive identifier in the Webble World display forms. Webbles can also be identified by a custom name set by the user. Such a name will be stored when saved, but it is not unique since many Webbles may have the same name. Another way to identify a Webble is by type name set when the Webble was published.

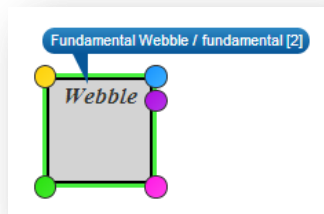


Figure 19: Displayed name consists of the user customized name, the unique id and the Webble type name.

When a Webble is found in the Webble browser it is identified by its type name. But that name is not useful when dealing with many loaded Webbles inside the Webble World desktop of the same type, since it is not unique. A user should identify his or her Webbles with a user defined name, but it is up to the user if he/she wants to keep those names unique when duplicating Webbles.

### Webble Menu

The Webble menu is the popup menu that each Webble carries with it and which is reachable either by right click with the mouse somewhere on the Webble or via the top left yellow Interaction Object on the Webble when selected. The Webble menu may be configured and even turned off though so the user should not be completely surprised if the menu looks and behaves different than expected.

## Webble Prototype

See Webble Template.

## Webble Template

Webble template or Webble prototype is also known as Webble type with a specific behavior. A Webble template has not been modified in any way from its default state. And is made with HTML, CSS and JavaScript only.

## Webble World Work Surface

The Webble World Work Surface is the actual work area where Webbles may exist and be manipulated, below the main menu.

## Workspace

A registered user may have one or many workspaces where he work and interact with the Webbles of his choice. Setups and configurations that he/she want quick access to. Workspaces in Webble World are separated work surfaces that a Webble user can use as needed. A workspace is private by default but may be shared with other user if wanted.

## Webble World System Menu

This section explains what each item in the top main menu does and why. The top main menu is open by default but may be closed either with Alt+F2 on computers with keyboards or with a quick swipe up on the work surface on touch devices. To open again is the same but swipe down instead.

### Workspace ->New

Cleans away all current Webbles on the current workspace and resets the internal Webble cache. The action is prompted with a confirmation since it cannot be undone.

### Workspace ->Open

Opens a popup form which lets the user select one of his/her stored workspaces to open and interact with. If Webble world is closed while a Workspace is open, that workspace will be restored automatically in its last saved state when the user returns for easy continuance.

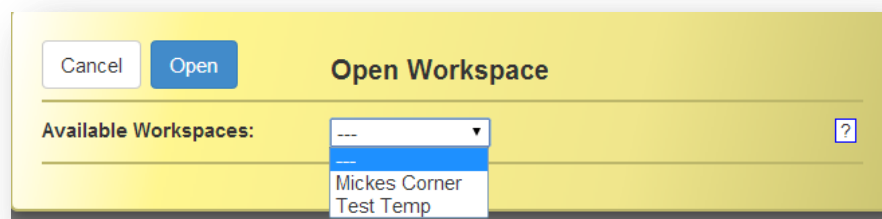


Figure 20: Available Workspaces are either those oneself saved previously or one that someone shared with the user.

The current open workspace also has a hotlink that is shown in the browser address bar and can of course be copy and pasted to anyone and anywhere.

### Workspace ->Save

Saves the current workspace under its current name or if it has no name prompts the user to give one and then saves it.

### Workspace ->Save As

Opens a popup form which lets the user select a name for the workspace to be saved and when submitted will save the current Webbles on the work surface inside a workspace with the chosen name

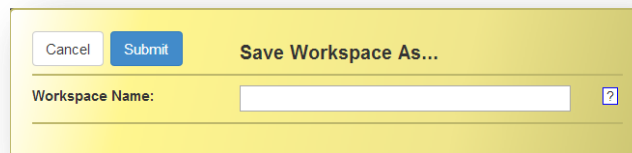


Figure 21: Any name works, as long as it is not used before. The system will help you if you are not sure.

### Workspace ->Share

Share Workspace means that you allow other selected users to edit and change it at their own will. This is especially useful when working together with other people on creating a Webble application or similar, but of course be used for other reasons like for example as a play area or game arena where each shared user is a player in an ongoing game. To share a Workspace one must know the user's username, at least well enough for the system to find and identify it.

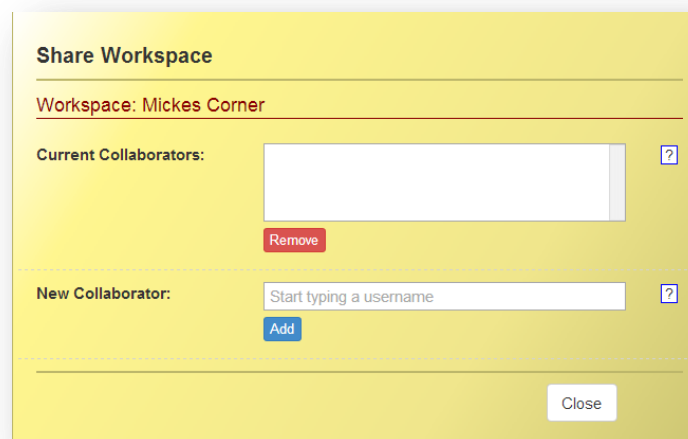


Figure 22: One does also remove already added collaborators within this form by select user and click remove.

If the user does not have a Webble World account or proper username, one can instead type an e-mail address which the system will use instead to contact the requested collaborator.

### Workspace ->Delete

This deletes the current open workspace from the server, never to be accessed again. Since it is quite a finite option the user is prompted to confirm the action. If no workspace is currently open this option does nothing.

### Workspace ->Print

Opens the browser built-in print dialogue in order to allow the user to make a print-out of the current visible work surface either on paper or pdf or any other available options the user's computer may provide.

### Webbles->Browser

Opens the Webble Browser (Figure 17) and lets the user search for all Webbles available in this Webble World System. The user can find Webbles by name, developer, keywords or ratings (including description text) and sort the result on the same. When the result of the search is displayed then a Webble may be selected and by clicking 'Load' (or drag & drop Webble image, or double click) the Webble will be picked up from the Internet and loaded to the Webble desktop. The Webble browser stays open for the convenience of a user who wants to pick more Webbles to load. When the user is finished he clicks 'Close' to close the Webble browser or click outside the form area and returns to the Webble desktop which should now contain the loaded Webbles.

When one selects a Webble a hotlink for that particular Webble is created in the browser address bar which can of course be copy and pasted to anyone and anywhere for quick loading the Webble in question without browsing for it.

The padlock icon before the Webble name (either locked or open) indicates weather this Webble is developed by a group you have explicitly assigned your trust or not. (See 'Username-> Profile Settings ->Trust' for more information on that).

Within the Webble browser is also found the possibility to rate individual Webbles and leave comments about them, both for helping other users to pick most suitable Webbles for their needs but also for making developers aware of what user likes and dislikes with the Webble in question.

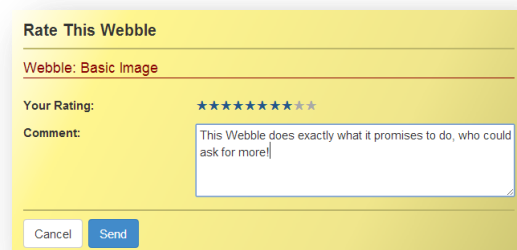


Figure 23: While the user base is still kind of small, many Webbles might still be unrated. Maybe you can help out with that.

## WEBBLE WORLD

### A FEW AVAILABLE DEFAULT WEBBLES [BROUGHT TO YOU BY MEME MEDIA LAB, HOKKAIDO UNIVERSITY]

#### Basic Button

A primitive Button Webble with basic button features.

#### Basic Image

A primitive Image Webble of the simplest fundamental kind.

#### Basic Label Webble

A primitive Webble that may be used to create a text label with many visually customizable slots.

#### Basic Rotate Needle Webble

A primitive rotating pin Webble that may be used for simulated analogue measurements of all sorts.

#### Basic Slider Webble

A primitive slider Webble that allows the user to manipulate numeral values with a slider mechanism.

#### Basic Text Input Webble

A primitive text input Webble with some visually customizable slots.

#### Basic Timer Webble

A primitive Webble that keeps track of the current time of the client system and can be used to collect time values. No internal visualization.

#### Clock

A compound Webble made up from a Basic Timer, Basic Image and three Basic Rotate Arms to create an ordinary simple analogue clock.

#### Fundamental

The simplest primitive Webble imaginable with only the most fundamental Webble features.

#### Random Number Generator

A Webble that can generate a random number between a specified min and max value.

---

### Webbles ->Load

To load a Webble manually you either need to know its exact location on the Internet, meaning the URL of the JSON file that describes it, or you browse for a Webble JSON file in the file system of your local computer.

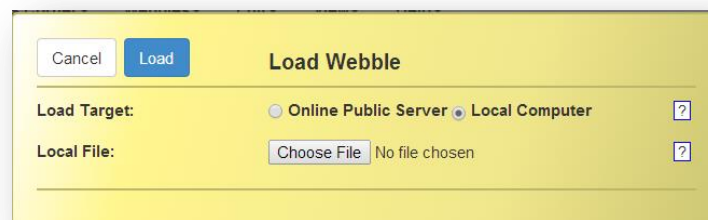


Figure 24: Any Webble online or locally can be loaded this way.

This is useful especially for Webbles that are not published in the Webble World system but still exists somewhere online or offline. A Webble loaded like this can still be saved and published in Webble World and then found via the Webble browser.

#### Webbles ->Recent

Allows the user to quickly load the last loaded Webble without having to browse for it again in the Webble browser.

#### Webbles ->Publish

After a Webble Designer has finished his creation and Webble configuration she can publish it and make it known to the outside world via this menu item (Figure 18). In the form that opens, the user adds an internal name for the Webble as well as descriptive information about the creation. An auto generated image is created for the Webble but if it is not cool enough it can be replaced with a custom image. If a Webble with the same name already exists the and that was not created by the current user, one will be informed about that and asked to choose another name, if a Webble with the same name exists and it's template was created by this user then a new and updated version of this Webble will created instead, leaving the old version intact though only available in Webbles which already use the previous version.

#### Webbles ->Template Editor

In the Template Editor an advanced Webble user with basic programming skills can create new Webble templates, either by uploading template code files written locally or by generate them directly in the Template Editor and write the needed code on site. The code required is a combination of HTML, CSS and JavaScript and how it is written is either explained in the comments of the auto generated files, or in the downloadable developer's package and/or in the last section of this document. When a Webble and its files have been generated in the editor the Webble can then be seen and interacted with.

Figure 25: In this area you upload or auto-generate and edit html5 files you have created to build a Webble Template and additional information about it. It will then appear in the users sandbox of hidden Webbles, where it will stay until published.



This area is also known as the sandbox for it is like a play-ground where a developer can experiment and test his/her creations. If a Webble is given the same name as a previous uploaded and published Webble, created by you, it will be stored online as a new version of the same. You can only upload Webble templates if you are a registered Webble World User and logged in.

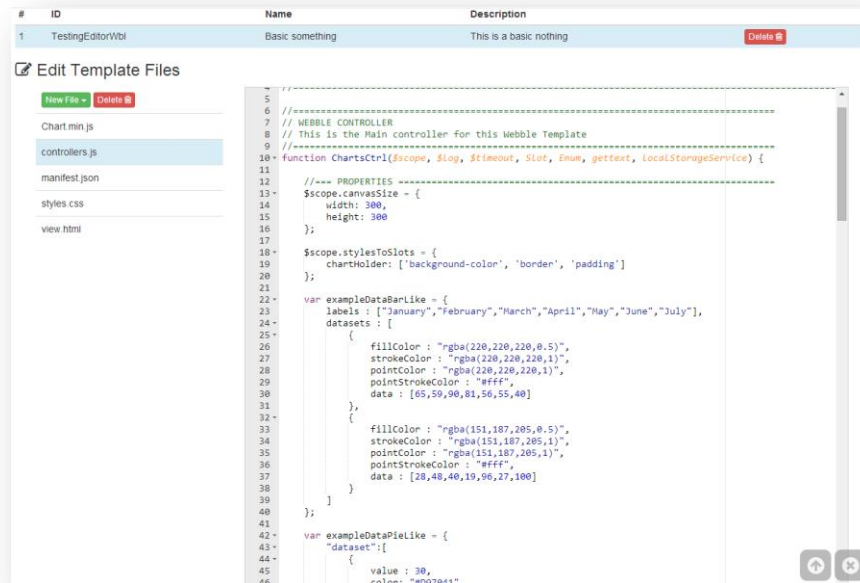


Figure 26: The developer can edit the code directly in the browser and immediately see the result and possible debug information when loading the Webble into the environment.

One can also type in the existing template id of a known Webble and by doing that be offered to get a copy of that template with all its code to the sandbox for creating a new template based on the previous one. This is a very powerful way to improve and build upon others peoples initial work. (Same thing can be achieved via a Webbles About form)

#### **Webbles ->Sandbox Webbles (listed by their names)**

Webbles uploaded into the sandbox via the previously mentioned form are not available via the Webble Browser since they are not yet published, but are instead listed in the Webbles menu for easy and quick access.

#### **Edit ->Undo**

Every major Webble change is saved so that you can go back via this menu item if you do something you regret, but one should be aware that many changes are happening in the background so when stepping far one might have to go further than expected in order to reach the goal. Undo works best when just going one or two steps backwards and it does not work well with animated Webbles which slots change multiple times each seconds for obvious reasons.

### Edit ->Redo

If the user made an undo that the user regrets and want to redo again, this is where one would perform that action. See Edit->Undo for further details about undo/redo in Webble World environment.

### Edit ->Select All

This selects all Webbles on the Webble World work surface (even the internal ones inside a bundle).

### Edit ->Deselect All

This deselects all Webbles on the Webble World work surface (even the internal ones inside a bundle).

### Edit ->Duplicate

This creates a duplicate of all selected Webbles. If no Webble is selected no duplication is made.

### Edit -> Shared Model Duplicate

Makes a duplicate of all selected Webbles, but instead of duplicating the model element, represented by all slots, as well, the new Webble will share the same model element with its original, meaning that all slots values are shared with its duplicate except for positioning slots which by default are decoupled, but that can be changed inside the property form which for Shared Model Webbles displays an extra column of checkboxes that allow the user to select which slots this Webble should share with its shared model duplicates. Shared model duplicates have a dashed border when selected so that they are easily spotted.

Internal Slots: Internal			
Display Name:	Fundamental Webble		?
Slots: fundamental			
Message:	Webble	<input checked="" type="checkbox"/>	?
Slots: CSS Related (fundamental)			
Width:	70px	<input checked="" type="checkbox"/>	?
Height:	70px	<input checked="" type="checkbox"/>	?
Background Color:	#D3D3D3	<input checked="" type="checkbox"/>	?
Border:	1.82px solid rgb(0, 0, 0)	<input checked="" type="checkbox"/>	?
Border Radius:	5px	<input checked="" type="checkbox"/>	?
Font Size:	15.45px	<input checked="" type="checkbox"/>	?
Foreground Color:	#333333	<input checked="" type="checkbox"/>	?

Figure 27: check or uncheck the slots that should be shared between shared model duplicates.

**Edit ->Bundle**

Bundle means wrapping all selected Webbles inside a bundle-Webble which will create a customized surface for the containing Webbles with only selected slots to be visible from the inside while at the same time shield the user from having to deal with any of the internal complexity of the Webble pack. When the user clicks Bundle all selected Webbles (and their children) are being Bundled together with a Bundle parent, and a form opens that allows the user to pick which slots from all the internal Webbles will be accessible from the bundle.

Figure 28: The Bundle Webble will also have some of its own slots for visual configuration outside the ones the user pick.

When the user have selected which (if any) and clicked submit the bundle is finalized, and one can see that by the light thin border that surrounds the containing Webbles. The Bundle Webble can be edited, combined and configured like any other Webble.

**Edit ->Delete**

Deletes all selected Webbles and their children.

**Edit ->Multi-Webble Properties**

Opens a form that allows the user to edit the properties (slots) of one or multiple Webbles at once. The form is not modal so the user can continue to interact with the Webbles while keeping the form open. Any change to the values in the form will not have immediately effect on the selected Webbles. Instead the user will have, by its own will, to push the changes to the selected Webbles. The same works for extracting changes done via other interactions so that they get mirrored in the multi-Webble form.

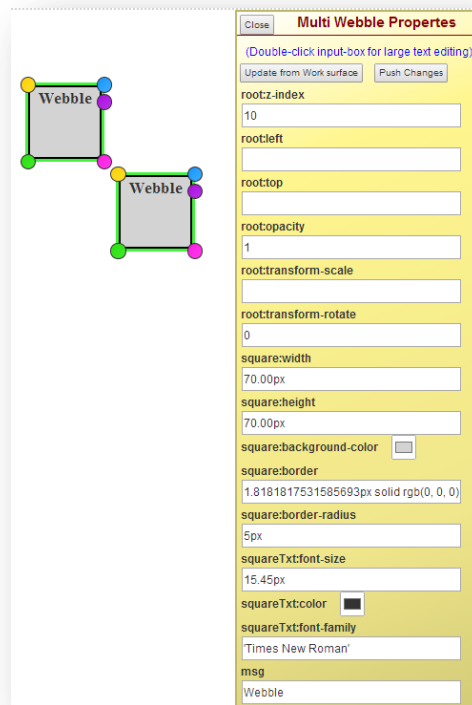


Figure 29: Webble property value manipulation for one or multiple Webbles.

If Webbles of different types are selected together only properties (slots) that share the same name are displayed.

#### Edit ->Platform Properties

Opens a System Property form, which allows the user to change the background color of the Webble work surface along with the current execution mode of the system.



Figure 30: Future versions of the Webble World system will probably provide a larger range of properties.

Auto Behavior is by default on but can be turned off from here, meaning for example that Webbles will not automatically create any default slot communication.

Some Webbles (though rare) may contain code that overrides these settings for optimal display of those Webbles.

#### View->Toggle Connection View

When shown the connection view displays a layer that indicates how Webbles are related with each other including shared duplicates and also what kind of slot connection that is currently running. This feature is for development and debug purpose mainly. The information is shown as a line for parent-child connection with arrows to indicate the directional flow of slot communication. Which slot is displayed as well. A colored circle with a number shows which Webbles share slots with each other.

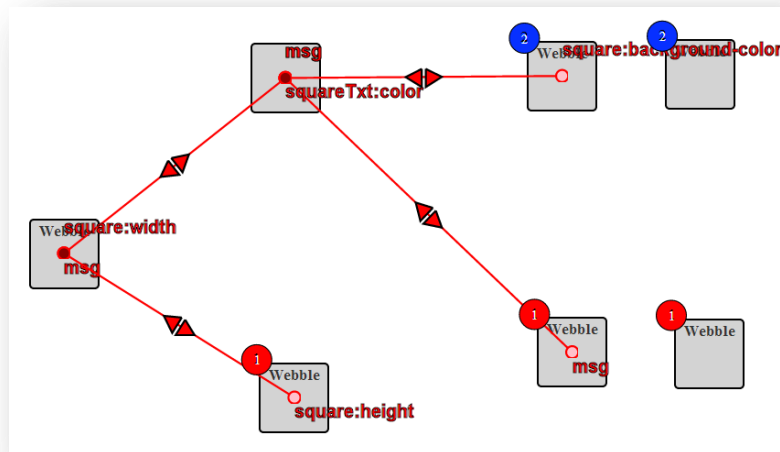


Figure 31: The Connection View layer, will automatically hide again when the user starts interacting with the Webbles.

#### View ->Active Workspace info

This displays an information box about the current open workspace. The information displayed is how long the user have been using Webble World during this session, the total amount of Webbles that are displayed, how many templates (prototypes) that has thus far been loaded into the system and also the amount of different Webble Types that are currently used in this workspace.

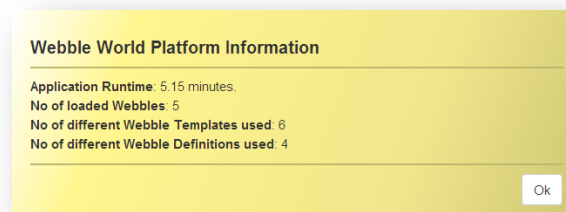


Figure 32: Saves you from counting the Webbles by yourself.

**View ->Toggle Full Screen**

Toggles full-screen mode for the Webble World Desktop.

**Help->Documentation**

Let the user download this manual and read it.

**Help->FAQ**

If a user has any questions about the system, then often the answer may be found inside the FAQ. If the answer is not found the user can post a new question, which will be answered by admin staff as soon as possible.



Figure 33: The FAQ section is probably the best and primary attempt to get answers on general questions about Webble World.

**Help->Open Chat**

The Webble World chat is a completely open non-private chat which anyone using Webble World can see and write in. Its main existence is for quick interaction between Webble World Users (even unregistered ones) and begins the first phase in forming a strong and supportive Webble World community.

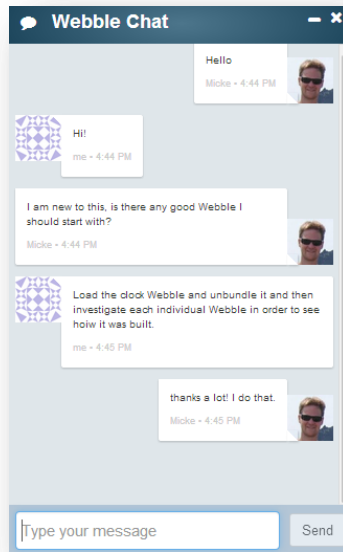


Figure 34: The Webble chat is yet another way of getting in touch with other Webble World users around the world.

Since it is a completely open chat, one should of course never reveal any private or sensitive information on this channel.

#### **Help ->Support**

Let the user mail (using the local machine installed mail program) the Webble World developer team with specific and detailed technical issues of a kind that not is suitable in the FAQ.

#### **Help ->Developer Community**

This is just a link to the Stack Overflow Website where Webble developer may ask each other questions on how to best develop your Webble template .

#### **Help ->Download Developers Package**

Let the user download necessary and supportive files and templates for beginning develop Webbles by code (HTML, JavaScript and CSS) on his/her own computer. The pack includes an example Webble, an extensive readme document and some reference code. In order to use the content, one only need a text-editor of any kind. (for example the built in Template Editor)

#### **Help ->Webble Platform on GitHub**

This menu item takes you to GitHub where you find the core code of the whole Webble World Platform. For ordinary Webble developers and users this is most likely of no interest, but for those who wish to improve upon the core code or understand the underlying system better or even start developing your own Webble World then this are of the Web might be of some interest.

### Help ->Report Bug

Let the user mail (using the local machine installed mail program) the Webble World developer team, informing them of bug issues in the system. This is only for bugs in the Webble World platform that more or less affects every Webble, and not for bugs in single Webbles which were not developed by private users. Those bugs are currently best addressed via the rating system.

### Help ->About

This displays an information box about Webble World 3.1 and its development team.

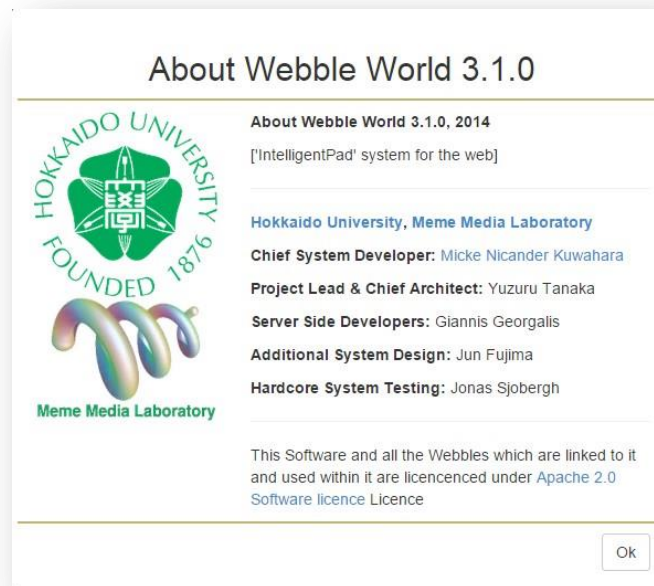
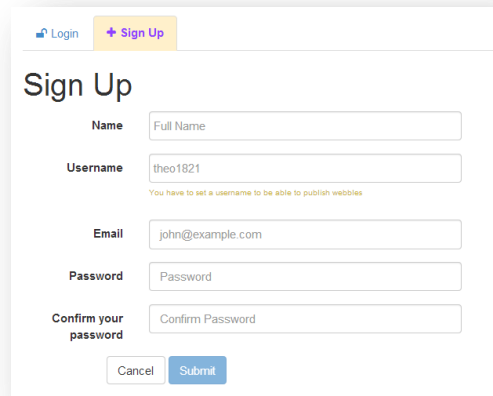


Figure 35: Who made this extraordinary piece of software?

### Sign Up

Visible only if there is no user currently logged. Mainly used for creating an account within Webble World that the user does not want to be associated with any other social network account such like google, Twitter or Facebook.





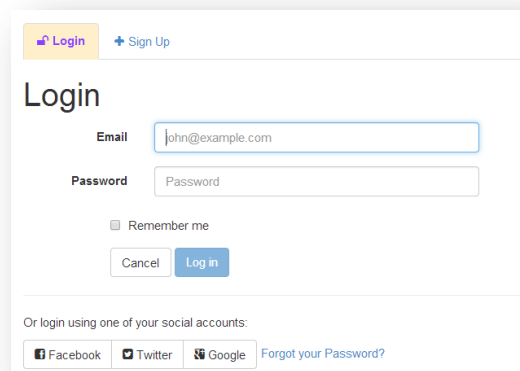
The 'Sign Up' form is a white rectangular box with a subtle shadow. At the top left, there are two links: 'Login' with a user icon and 'Sign Up' with a plus icon. The title 'Sign Up' is centered below the links. The form contains five input fields: 'Name' (placeholder 'Full Name'), 'Username' (placeholder 'theo1821' with a small note below: 'You have to set a username to be able to publish webbles'), 'Email' (placeholder 'john@example.com'), 'Password' (placeholder 'Password'), and 'Confirm your password' (placeholder 'Confirm Password'). At the bottom are 'Cancel' and 'Submit' buttons.

Figure 36: Type in the requested information and you are one step closer to publishing your own Webbles.

This is not the fastest and easiest way to create an account with Webble World, for that look below at the 'Login' section.

### Login

Here the user can either login to an existing account or create an account by associating it to an existing account in the users social network. Clicking for example the google button will either logged in a user who is already registered and has connected its google account to the Webble World account or create a new account and automatically connect it to the chose social network account.



The 'Login' form is a white rectangular box with a subtle shadow. At the top left, there are two links: 'Login' with a user icon and 'Sign Up' with a plus icon. The title 'Login' is centered below the links. The form contains two input fields: 'Email' (placeholder 'john@example.com') and 'Password' (placeholder 'Password'). Below the password field is a 'Remember me' checkbox. At the bottom are 'Cancel' and 'Log in' buttons. Below the login fields, there is a section titled 'Or login using one of your social accounts:' with three buttons: 'Facebook', 'Twitter', and 'Google'. A link 'Forgot your Password?' is also present.

Figure 37: After logged in the user is automatically logged in again from the same device and browser when returning to Webble World.

After being logged in the Sign Up and Login menu items disappears and are replaced by the username and a user specific drop down menu.

### Username-> Profile Settings->Account Information

Here the user can edit and add information about his user account such as username (if such has not yet been set, if it has been set it cannot be changed), preferred Webble World platform language when logged in, user description etc.

The screenshot shows the 'Account Information' section of the Webble World user interface. It includes input fields for 'Name', 'Email', and 'Website'. A 'Preferred Language' dropdown menu is set to 'English'. There is a 'Birthday' field with a calendar icon, currently showing 'Unknown'. A 'Gender' section has radio buttons for 'Male' and 'Female'. A 'Description' text area is at the bottom. A blue 'Save' button is located at the bottom center of the form.

Figure 38: Currently all this information is private and not displayed to anyone else, except username.

### Username-> Profile Settings ->Notifications

This area is not yet fully active but when it is activated it gives the option to receive e-mail information about Webble World news and related. But some specific actions is noted here when they occur.

The screenshot displays the 'Notifications' settings page. It features three notification preferences, each with an unchecked checkbox: 'Get an email every time the platform has major updates', 'Get an email every time someone else publishes a new version of a webble of which you are coauthor', and 'Get an email every time someone shares a workspace or a webble with you'. Below these is a section titled 'Unhandled Notifications' with the instruction 'Please review and respond to the notifications below.' and a message box stating 'There are not any unhandled notifications -P'.

Figure 39: If your user is assigned to a group a notification will appear here.

### Username-> Profile Settings ->Trust

Since Webbles can be very powerful and contain a lot of features we do not always know about or want, Webble World also comes with a system of trust. Where a user gets the informed about every Webble's origin and weather it belongs to a development group that the user trusts or not. In this section the user can assign trust to groups he/she

belongs to as well as groups he feels comfortable with. In the Webble browser the user gets informed via a padlock icon about each Webbles level of trust based on the user's preferences.

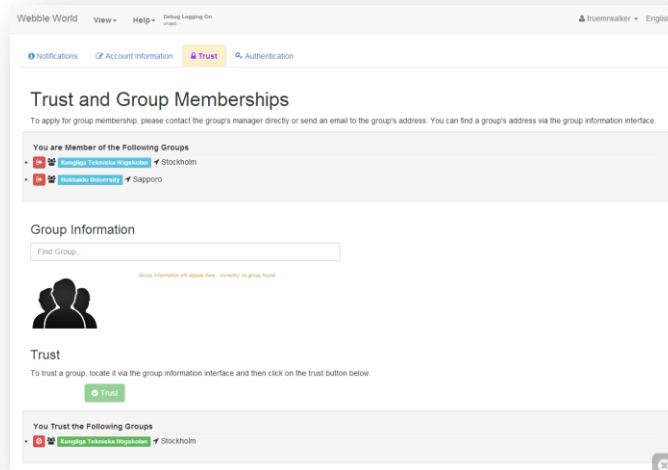


Figure 40: Here you assign or revoke your trust to groups existing in Webble World and the Webbles they develop.

#### Username-> Profile Settings ->Authentication

Here the user can connect the Webble World account to social media sites, such as google+, Twitter and Facebook. It is also here one would change the Webble World password, which is only used when the user does not log in via any of the above mentioned social media accounts though.

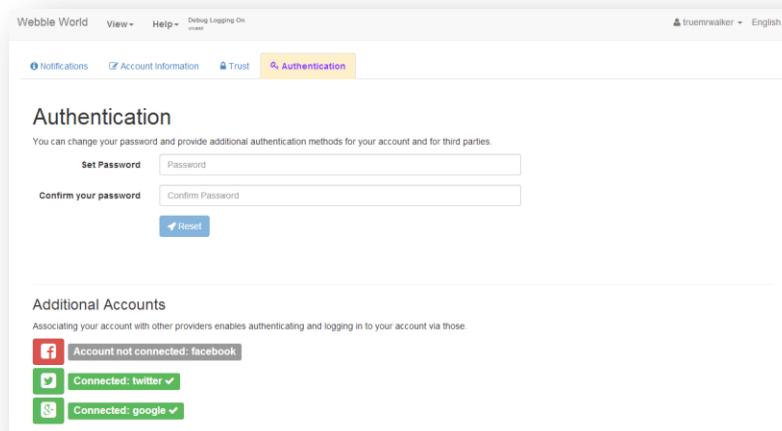


Figure 41: More associative social media sites may become available in the future.

It is also here the user would register any API keys he or she intend to use when loading or developing Webbles that require such. Because API keys are personal but Webbles are shared, Webble World offers users to use personal API keys instead of forcing a developer to add his/her keys inside the code of the Webble. How to apply to keys is not covered in this document and how to access these registered keys in a Webble can be found in the template comments in the developer's package. API keys can also be registered by a group and thereby be available by all members of that group without the need for personal keys.



**Licenses and API keys**

You can manage your personal licenses and API keys via the interface below.

**Realm**

**Resource**

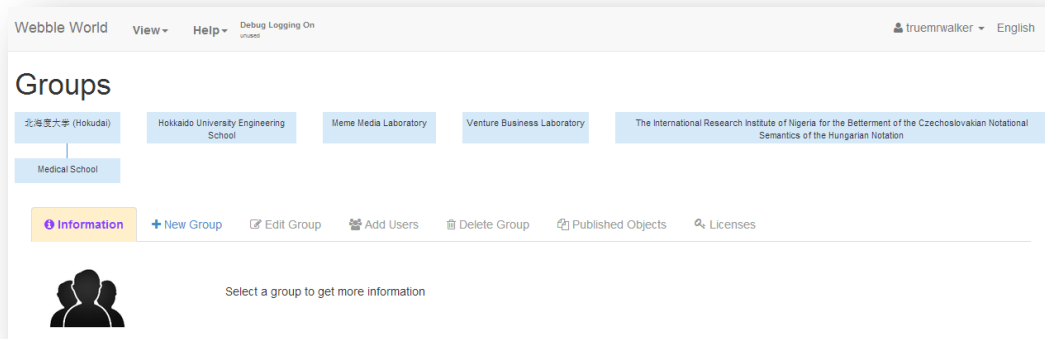
**Access Key**

[+ Add](#)

Figure 42: Most Webbles probably do not require these keys, but those who do, now have a very good way of how to deal with that.

### Username-> Groups

This area displays information on all registered groups that develops Webbles for Webble World. If you are registered as the role of an administrator or a user who have been assigned ownership of a group by an administrator more features is available like edit group info, create sub groups and assign users to groups.



Webble World View Help Debug Logging On

truemwalker English

## Groups

北海道大学 (Hokudai) Hokkaido University Engineering School Meme Media Laboratory Venture Business Laboratory The International Research Institute of Nigeria for the Betterment of the Czechoslovakian Notational Semantics of the Hungarian Notation

Medical School

[Information](#) [+ New Group](#) [Edit Group](#) [Add Users](#) [Delete Group](#) [Published Objects](#) [Licenses](#)


 Select a group to get more information

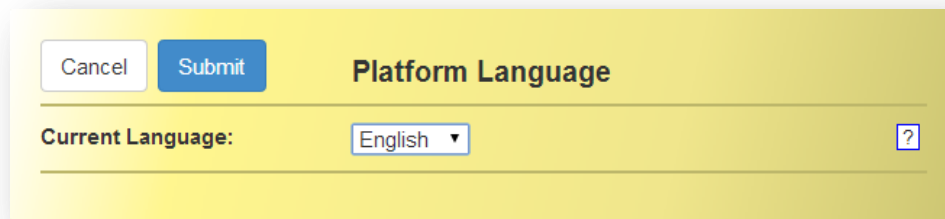
Figure 43: Groups in Webble World is a way of making connections and develop trust.

**Username->Logout**

If the user does not want to be logged in anymore then he or she can log out here. Automatic login will then also be canceled.

**Language**

If the user has not set the preferable language in the profile or if the user does not yet have an account or are logged in, then this option allows changing the current language. Language changes outside of the user profile are not remembered by the system between sessions.



The image shows a yellow dialog box titled "Platform Language". At the top left are two buttons: "Cancel" and "Submit". Below the title bar, there is a label "Current Language:" followed by a dropdown menu showing "English" and a small blue square icon with a white question mark to its right.

Figure 44: Languages available increase by time, so if your language is not present yet, please be patient.

**Default Webble Interaction Balls**

When a Webble is selected by a user, a bounding box, a frame, appears and surrounds the Webble. Along the path of that frame, interaction points have been distributed in the shape of small balls which will grow in size when activated. These balls can be used for any mouse interaction, such as click, drag and drop. Five of these balls are set up by default and are visible from the start, though Webble developers and Webble users may add more balls or even hide the default ones on their own created or manipulated Webbles. Here follows a description of the four default Interaction Balls and how they are used.

**Open Menu**

*Color: Yellow*

*Position: Top Left*

*Interaction Type: Click*

Opens the selected Webble's pop-up menu so that the user may choose an action

**Rotate**

*Color: Light blue*

*Position: Top Right*

*Interaction Type: Drag*

Rotates the Webble from its center using its angle slot value. If the Webble has children and grandchildren, those will also rotate around the rotating center. The rotation may be negative as well.

### **Resize**

*Color: Pink*

*Position: Bottom Right*

*Interaction Type: Drag*

This interaction object is not always present, that depends on the construction of the Webble weather it has a proper Width and Height value to attach to. If it does the Webble will resize in width and height when dragging the mouse horizontally and vertically. This action only affects the Webble being resized and not any children or grandchildren.

### **Assign/Revoke Parent**

*Color: Green*

*Position: Bottom Left*

*Interaction Type: Click*

Same as 'Assign Parent' from the popup menu described below, meaning allowing the user to select another Webble to become the chosen one's parent. It will not be visible if the Webble already has a parent.

## **Webble Default Popup menu**

When you right click with the mouse on a Webble or when the yellow top left Interaction Ball is clicked the Webble menu is displayed. Below follows a description of the default menu items that come with a Webble. Though a specific Webble may have many extra custom menu items and may even have the default ones disabled, the user is most likely to find the items below at some stage during 'Webbling'.

### **Publish**

Same as *Webbles->Publish* in the main menu but with the targeted Webble (and its children) as the only target no matter how many other Webbles might be selected.

### **Duplicate**

Same as *Edit->Duplicate* in the main menu but with the targeted Webble (and its children) as the only target no matter how many other Webbles might be selected.

### **Delete**

Same as *Edit->Delete* in the main menu but with the targeted Webble (and its children) as the only target no matter how many other Webbles might be selected.

### **Assign Parent**

Lets the user pick a Parent for the Webble by clicking on the Webble wanted. When this menu option is selected the target Webble gets a golden glowing border to indicate it is the future child and all other Webbles that might be able to become a possible parent gets a light blue glowing border. These glowing borders provide a visualization of the process. After the user have selected a parent by clicking or tapping on it the border of the child glows light pink while the selected parent's border glows in deep red. Non-selected Webbles goes back with no border. This menu target is not visible if the Webble already has a parent. Same as using the green bottom left Interaction Ball.

**Revoke Parent**

Lets the user revoke the Parent connection for the Webble. This menu target is not visible if the Webble has no parent.

**Connect Slots**

Opens the slot connection manager (Figure 13) for a selected Child Webble. The slot connection manager lets the user pick two slots – one in the child and one in the parent – that are to exchange values with each other. In the manager the direction of communication may be set to send and/or receive or none. As soon as the connection is submitted the Webbles will start communicating their slot values, and will continue to do so until the slot connection is broken or changed by the user. A Webble that lose its parent will still remember its connection and try to establish it again automatically if the Webble is pasted to a new parent. This menu target is not visible if the Webble is not a child of another Webble.

**Properties**

Opens the Property manager (Figure 6) for the active Webble. The properties are customized for every type of Webble and range from just a few to up to a hundred or more. They are also divided into several logic groups. If there are any user generated custom slots for this Webble they will mainly show up at the bottom of the list.

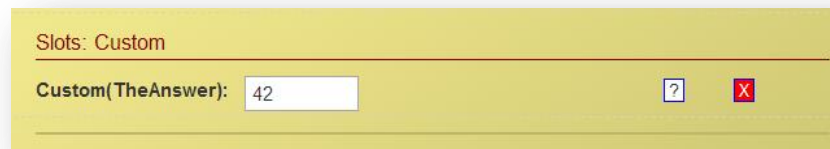


Figure 45: To delete a custom slot one just clicks the X.

The user may change all these properties freely but the range of available values is dependent on property type. Some values contain multiple CSS values in one, like is common in the style sheet world. The slot name is displayed with its translated language-dependent name, and with its internal slot name as a tooltip info when hovering the slot name. You can also click the slot name and then open up a small text prompt for quick saving of the slot name in the clipboard using Ctrl+C. The slot value visible is the current value, and if one hovers the mouse pointer above the question mark detailed information about the slot is displayed.

## AVAILABLE VISIBLE PROPERTY TYPES

**Text**

Plain text – Write whatever

**Numeral**

Any number, with or without a decimal point, signed or unsigned. Only numerical values are allowed; everything else is ignored.

Check Box

True or false check mark

Color Pick

Click the colored square to open a color selector, select a color and click the small OK button to confirm the selection; the color will change in the square. Or write the color by hand using its hexadecimal color code.

Combo Box

Drop-down list where the user may select one option

Date Pick

Calendar drop down box that allows the user to select a date.

File Browser

text field with a browse button that allows the user to locate and pick a file on the local computer.

Fonts

A drop-down list with fonts displayed as they will occur when selected. The user may pick one. Alternatively a textbox where the user can type font name as free text,, in this case multiple.

List Box

A list of displayed values from which the user may select one or many options.

Radio Button

A group of options where one is selectable via a traditional radio button group

Image Pick

Click the Choose File button to open a select image dialogue, where the user can pick an image from the local computer, which will be saved as image data, or by typing a URL in the textbox. The image will change accordingly. Only an absolute path is usable unless the image is stored inside the Webble folder, in which case a relative path works too. If one has copy pasted proper web image data from somewhere else that can also be pasted in the textbox and used.

Media Pick

Works similar to Image Pick but for .mp4 and mp3 files.

Rich Text

Text that can be decorated and comes with a small additional WYSIWYG editor for HTML and CSS editing

Slider

Move a slider left to right and back again to choose a ranged numeral value. In some cases the change is displayed in a square beside the slider.

Password

Text field where the text is hidden and replaced by stars or similar.

Private

Slot displayed but not editable. Sometimes the value is hidden and only the name visible.

Point

Two numerical values that represent an X and a Y position of some kind

Border

Four numerical values that represent a left, right, top, bottom group of some kind



When all changes are made in the property manager the user clicks ‘Submit’ and all changes will be imposed on the Webble. If ‘Close’ is clicked all un-submitted changes are disregarded and the form will close.

#### Bundle

Same as *Edit->Bundle* in the main menu but with the targeted Webble (and its children) as the only target no matter how many other Webbles might be selected.

#### Unbundle

Unbundles a bundle-Webble and release all internal Webbles to be fully interacted with again.

#### Shared Model Duplicate

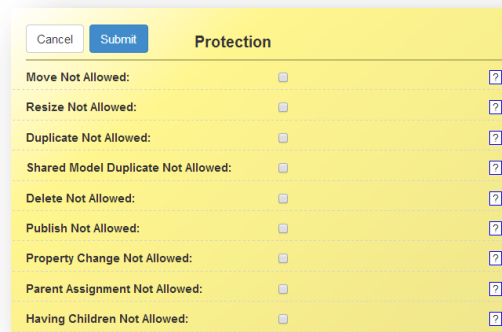
Same as *Edit->Shared Model Duplicate* in the main menu but with the targeted Webble as the only target no matter how many other Webbles might be selected. Note that this does not make children sharing their slots and model.

#### Bring To Front

Sets the layer order to the highest among all Webbles and make it therefore be on top of all other, as long as they are within the same level in the parent child hierarchy. A parent Webble cannot float above a child. Layer Z index slot may also be changed via the Property manager.

#### Set Protection

If a user wants to protect a Webble from Some operations, that can be managed from here.



Protection		
Move Not Allowed:	<input type="checkbox"/>	?
Resize Not Allowed:	<input type="checkbox"/>	?
Duplicate Not Allowed:	<input type="checkbox"/>	?
Shared Model Duplicate Not Allowed:	<input type="checkbox"/>	?
Delete Not Allowed:	<input type="checkbox"/>	?
Publish Not Allowed:	<input type="checkbox"/>	?
Property Change Not Allowed:	<input type="checkbox"/>	?
Parent Assignment Not Allowed:	<input type="checkbox"/>	?
Having Children Not Allowed:	<input type="checkbox"/>	?

Figure 46: The user checkmarks the things this Webble should not be allowed to do.

A protection makes it impossible to oppose an operation of the protected kind on the Webble being protected. Many of these protections may be overridden when the user is in Developers execution mode and holding Alt key (sometimes also Shift key).

**AVAILABLE PROTECTIONS**

Move Not Allowed

The Webble cannot be moved from its current location

Resize Not Allowed

The Webble cannot be scaled

Duplicate Not Allowed

The Webble cannot be duplicated (but shared duplicate is okay)

Shared duplicate Not Allowed

The Webble cannot be shared duplicated (but ordinary duplicate is okay)

Delete Not Allowed

The Webble cannot be deleted

Publish not allowed

The Webble cannot be published, meaning that any other Webble connected with it cannot be published either

Property Change Not Allowed

The Webble cannot change any slots or other properties

Parent Assignment Not Allowed

The Webble cannot assign any parent to itself

Having Children Not Allowed

The Webble cannot be assigned as a parent to any Webble

Parent Revoking Not Allowed

The Webble cannot revoke any parent from itself

Disconnect Children Not Allowed

The Webble cannot be revoked as parent

Bundle Not Allowed

The Webble cannot be bundled

Unbundle Not Allowed

The Webble cannot be unbundled.

Default Menu Not Visible

All the default options in the Webble popup menu are removed and disabled.

Interaction Objects Not Visible

Never displays the interaction objects of the Webble, though the Surrounding border will still be visible when selected.

Selection Not Allowed

The Webble cannot be selected (No interaction balls or selection border are displayed)

Context Menu Not Allowed

The popup menu never appears.

Non-Developers Mode Not Visible

The Webble completely disappears from view when not in Developers execution mode.

#### Drag Clone Not Allowed

When this is checked then the Webble will not use a dragging clone (ghost) when being dragged but instead move the actual Webble and therefore update the position slots in real time.

#### Bundle Locking Not Allowed

When this is checked then the Webble can still be individually selected and dragged even though it is a part of a bundle.

### Add Custom Slot

Via this menu item the user can add custom slots to the Webble. It allows three types of custom slots to be created.

Figure 47: Adding custom slot can be immensely helpful when building complex Webbles.

First the classic custom slot of any common type like text, Boolean, numbers etc. which only requires a name and any value that can be typed. The second is a slot which is connected to a specific CSS style sheet attribute that has to be selected by the user from the internals of the Webble via the form. The third is the option to create a merged slot that contains a set of other existing slots together as one for more complex slot communication. The user can remove custom added slots via the properties form.

When connecting merged slots the Slot Connection Form will change a bit and allow the user to specify which part of the merged slot is going to connect to which part of parent slot. This is done easily by a simple cross-section matrix.

CancelSubmit

Slot Connection

Child Slot:

Custom (myTestMerge) [myTestMerge]

?

Parent Slot:

Custom (myTestMerge) [myTestMerge]

?

Merged Combo Slot channels:

?

Child \ Parent →	square.width	square.background-color	squareTxt.font-size
square.width	•		
square.background-color		•	
squareTxt.font-size			

Connection Directions:

☒ Send

☒ Receive

?

Figure 48: Just click in the cross section for the internal slots in the merged slot that you wish to share value.

Custom Menu Items

Via this menu item the user can edit the Webble menu, both by removing default menu items but also by adding new custom items which are assigned to make some form of slot changes.

CancelSubmit

Edit Custom Menu Items

Item Id	Item Display Name	Item Action	Item Enabled
Default Menu Items			?
Custom Menu Item			?
goBlue	Go Blue	<div>Change Slot</div> <div>Background Color [square.background-color]</div> <div>blue</div> <div>None</div> <div>Done</div>	<input checked="" type="checkbox"/>
		<div>None</div>	<input checked="" type="checkbox"/>

Figure 49: An easy way to get the Webble to do a complex set of slot changes with one click.

This is a good way to make your Webble more user-friendly and faster to operate. You can bind a range of slot changes to just one menu item.

Custom Interaction Objects

Via this menu item the user can edit the interaction objects, both by removing default objects but also by adding new custom interaction objects which are assigned to make some form of slot changes.

Cancel

Submit

Edit Custom Interaction Objects

Color	Index	Position	Name Id	Tooltip Text	Action	Mouse Event Type	Enabled
	0		<div>Menu</div>	<div>Open Menu</div>	<div>Default</div>	<div>Mouse Click</div>	<input checked="" type="checkbox"/>
	1		<div>Rotate</div>	<div>Rotate</div>	<div>Default</div>	<div>Mouse Move</div>	<input checked="" type="checkbox"/>
	2		<div>Resize</div>	<div>Resize</div>	<div>Default</div>	<div>Mouse Move</div>	<input checked="" type="checkbox"/>
	3		<div>AssignParent</div>	<div>Assign Parent</div>	<div>Default</div>	<div>Mouse Click</div>	<input checked="" type="checkbox"/>
	4		<div></div>	<div></div>	<div>None</div>	<div></div>	<input type="checkbox"/>
	5		<div>Rescale</div>	<div>Rescale</div>	<div>Default</div>	<div>Mouse Move</div>	<input checked="" type="checkbox"/>
	6		<div></div>	<div></div>	<div>None</div>	<div></div>	<input type="checkbox"/>
	7		<div></div>	<div></div>	<div>None</div>	<div></div>	<input type="checkbox"/>
	8		<div></div>	<div></div>	<div>None</div>	<div></div>	<input type="checkbox"/>
	9		<div></div>	<div></div>	<div>None</div>	<div></div>	<input type="checkbox"/>
	10		<div></div>	<div></div>	<div>None</div>	<div></div>	<input type="checkbox"/>
	11		<div></div>	<div></div>	<div>None</div>	<div></div>	<input type="checkbox"/>

Figure 50: The user have 12 interaction objects available to them to manipulate.

This is a good way to make your Webble more user-friendly and faster to operate. You can bind a range of slot changes to just one interaction object. The object can also manipulate the slot value by mouse movement.

About

This displays a pop-up info box that contains some specific data and information regarding the Webble in question. One major use of this area is the blue button next to template id that allows the user to create a copy of the template code of that template in order to use as a base for developing a new and hopefully better and more powerful Webble.

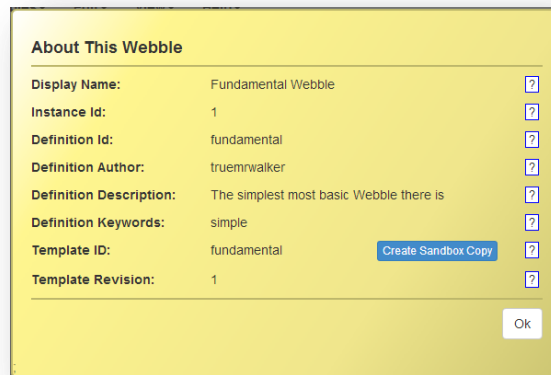


Figure 51: This information is mainly interesting when looking to identifying the author or id of a Webble or building a similar version of its template.

## Short-Cut Keys

Most menu items may also be reached via keyboard short cuts if the Webble World visit is performed on a computer, such as a PC or a Mac and if the user is currently running in Development execution mode. There are also some short-cut key only options useful for developers. Short-cuts does not work in any other execution mode.

### Esc

Close all forms and unselect all Webbles

### Enter

In the Webble Browser the Enter Key execute a search or load the selected Webble, and in some other forms the enter key executes a submit, though not in all.

### ALT + F1

Displays a popup box with information of what all non-menu related short-cut keys does.

### ALT + F2

Toggles the Webble World top menu visibility.

### ALT + F3

Toggles between logging and non-logging of debug activities inside Webbles and Webble World platform. It is up to Webble prototype developers weather such log writings is used or not, but if it is, this key will enable it.

### ALT + F5

Saves a local copy of the current work surface for fast retrieval at a later time.

### ALT + F6

Retrieves a local copy of a previously saved work surface is such exists, replacing the current one.

**ALT + F8**

Loads a fundamental Webble. This is for quick access when testing system and Webble features. .

**ALT + F9**

Automatically switch to the system language if that is not already set, and if that is set instead immediately change to English.

**ALT + F10**

Opens the language switching popup form.

**ALT + SHIFT + Page Down**

Clears the cookie that keep in memory that the intro movie has already been seen, meaning that next time you visit Webble World the intro movie will play again.

**ALT + SHIFT + END**

Clears all cookies and all locally stored user memory data.

**ALT + Arrow keys**

Moves all Selected Webbles in the direction of the key

**ALT + N**

Same as *Workspace->New* in the main menu

**ALT + O**

Same as *Workspace->Open* in the main menu

**ALT + S**

Same as *Workspace->Save* in the main menu

**ALT + J**

Same as *Workspace->Share* in the main menu

**ALT + X**

Same as *Workspace->Delete* in the main menu

**ALT + SHIFT + P**

Same as *Workspace->Print* in the main menu

**ALT + B**

Same as *Webbles->Browse* in the main menu

**ALT + L**

Same as *Webbles->Load* in the main menu

**ALT + R**

Same as *Webbles->Recent* in the main menu

**ALT + P**

Same as *Webbles->Publish* in the main menu

**ALT + U**

Same as *Webbles->Upload Template* in the main menu

**ALT+ Z or CTRL + Z**

Same as *Edit->Undo* in the main menu

**ALT+ Y or CTRL + Y**

Same as *Edit->Redo* in the main menu

**ALT + A**

Same as *Edit->Select All* in the main menu

**ALT + SHIFT + A**

Same as *Edit->Deselect All* in the main menu

**ALT + D**

Same as *Edit->Duplicate* in the main menu

**ALT + SHIFT + D**

Same as *Edit->Shared Model Duplicate* in the main menu

**ALT + Delete or CTRL + Delete**

Same as *Edit->Delete* in the main menu

**ALT + E**

Same as *Edit->Multi Webble Properties* in the main menu

**ALT + SHIFT + E**

Same as *Edit->Platform Properties* in the main menu

**ALT + Numpad 9**

Same as *View->Toggle Connection View*, in the main menu

**ALT + I**

Same as *View->Active Workspace Info*, in the main menu

**ALT + F**

Same as *View->Toggle full screen*, in the main menu

**ALT + M**

Same as *Help->Documentation*, in the main menu

**ALT + SHIFT + F**

Same as *Help->FAQ*, in the main menu

**ALT + C**

Same as *Help->Open Chat*, in the main menu

**ALT + SHIFT + C**

Same as *Help->Developer Community*, in the main menu



**ALT + H**

Same as *Help->Support*, in the main menu

**ALT + G**

Same as *Help->Download Developers package*, in the main menu

**ALT + SHIFT + G**

Same as *Help->Webble Platform On GitHub*, in the main menu

**ALT + Numpad 7**

Same as *Help->Report Bug*, in the main menu

**ALT + Numpad 5**

Same as *Help->About*, in the main menu

**ALT + Mouse / ALT + SHIFT + Mouse**

Allows the user to enable things that has been disabled in the Webble if at the same time the user is in Development execution mode.

## Mouse Interactions

The mouse (or the finger on touch screens) is the main Webble interaction tool and it allows you, in the default behavior state, to open the Webble menu by right clicking on it and by double clicking a Webble to select it, drag and drop and move it, and to use Interaction Balls. Webbles may be dragged and dropped anywhere on the Webble World Desktop; if that position is outside the view the user can use the scrollbars of the desktop to find them. If the Ctrl-key is held when releasing a Webble after a drag and drop operation it returns to the original position and ignores the movement.

Webbles may also have many of their own mouse behaviors, for example in the case of the 'Fundamental Webble' it has a mouse over behavior that make the text italic. Many other Webbles may have many other types of mouse interactions.

If a Webble by default cannot be selected (displaying the Interaction Balls) the user can usually fix that by clicking on the Webble with Ctrl key down, if the execution Mode is set to 'Developer'.

Sometimes info text boxes appear, and have to be interacted with or closed before any further Webble activity may continue. Some info boxes may also close automatically after a certain time, or close when the user moves the mouse.

## Touch Screen Gestures

As far as it is possible the touch screen tries to mimic mouse clicks by taps, hold and drag, double tap and so forth. Interaction Objects surrounding the Webble can also be managed with finger touch, but might require an additional tap sometime instead of mouse over and mouse leave behavior. Some new gestures have been added though where the mouse operation was too complex or where the operation was purely controlled by keyboard. Those gestures are the following:

**Swipe Down [on the work surface]**

This will open the top main menu if it is closed.

**Swipe Up [on the work surface]**

This will close the top main menu if it is open.

More touch screen specific interactions may be added in the future when such appear useful and/or necessary.

## The Webbles

Though there are many features in the Webble World platform itself, it is but a drop in the ocean compared to the full impact of all the numerous Webbles that already exist and which still waits to see the light of day. The power of the Webble World lies within the Webbles and they may contain anything a web developer may imagine. And it does not only stop there because it continues with how Webble users chose to combine and use those Webbles in order to create new compound Webble gadgets and Webble applications. And what kind of features and behaviors, looks and anticipated usage those Webbles have there is no way to know in advance, you just have to keep coming back to Webble World and be amazed.

## Summary

This chapter hopefully told you anything and everything you ever wanted to know about Webbles but were too afraid to ask. Feel free to return here whenever your memory fails you on what to do and when.

Also be aware of the fact that individual independent Webbles may have far more features than described in this chapter, and this chapter never had the scope to cover all the thousands and thousands of possible future features that Webble developers will give us. Therefore do not be afraid if a certain Webble does things a bit differently from how you have been taught; just go with the flow and adapt. Flexibility is the key for success in all areas of life.

Congratulations on your newly gained status as a 'Webbler'. Now it is time to do some serious 'Webbling'. Enjoy!



## Developer's Manual

*So you want to be one of those selected few that will contribute to the world of Webbles in a more hard-core way. You want to make and distribute your own Webbles, ready to take on tasks that no other Webble has ever taken on before. You want to help out, and do your share of hard work, to fill the world's collected resources of Webble Repositories. So you want to become a Webble Developer. Well, get on with it then...*

This chapter will in detail give you the information you need to be able to develop your own Webble elements, but it will assume that you already are a semi-experienced web developer with full knowledge of HTML5, JavaScript, CSS3 and a basic understanding of AngularJS framework. Even though this chapter will cover most grounds, we also recommend though, in addition to the information found in this manual that you also carefully study the code and the comments found in the template source code files that you will base your development on, to make you a true master Webble developer.

If you are not yet a skilled web programmer in the languages used in this manual, but still wish to become a Webble developer we recommend that you get hold of some good literature and tutorials on the subject and first learn the necessary basic skills. There are MANY books on the subject but we would claim that the main resource for HTML5 and AngularJS understanding can probably be found on the Web, at for example these sites: <http://www.w3schools.com/> and <http://angularjs.org/>

There are no special tool requirements for Webble development. Of course the simplest and most straight forward approach would be to use the built-in Webble World Template editor, but if you want to use some local favorite editor that is completely up to you. We would recommend that you use an editor that supports AngularJS IntelliSense though. But in the end it is all up to you how and where you write your code. The only thing needed for a first time developer is to download the Webble Developers Package and read up on its content, available via the Webble World Platform under Help in the menu.

Let's begin our journey by learning how the Webble World is structured...

## Webble Architecture

So let's now find out which pieces of this puzzle that you should be aware of and how you may use them while developing your own Webbles. For further hardcore knowledge in any area, please take a look at the source code and its comments.

### Webble Parts

A Webble is defined by two separate parts called the Model and the View.

The model is the logical area of a Webble, represented by JavaScript code within AngularJS structured controllers and directives. It has no knowledge of how the Webble looks like or what the View is capable of. A specific instance of a model may be represented by many views. There are four separate files in the Fundamental template provided in the Developers Package that together represent the model from an AngularJS point of view, and that is the controllers.js file, the directives.js, the filters.js and the services.js. More details on each of those files further down. Only the controllers.js is required, while the others are only optional and basically just make sense for advanced Webble with a lot of under-the-hood logic.

The View is the user interface and the visual part of the Webble. It is here all interaction and all visualization of a Webble's behavior should be stored. The View is represented by the view.html file and the styles.css, these contain no JavaScript at all and its content is clean from caring about what the model does or how it does it, but they are most likely connected the AngularJS way by using scope variables that binds the data in the controllers.js to the displayed values in the HTML. If your Webble needs its own filters, directives and services, tags and attributes to find those may appear in the view.html as well.

If you will be using any 3<sup>rd</sup> party libraries, then one more file will be of interest to you; the manifest.json file. It is in there you declare all external and additional files which you need to be included to be loaded. Either only by name if they reside at the root of the Webble or by full path if they are found elsewhere.

The order of which these files are created or not created does not matter, but the usual approach is probably to begin with the view.html and styles.css, followed by controllers.js. Then work with them in parallel until the Webbles seem finished. The need for directives, filters and services probably appear during the development, and sometimes not at all. Same goes for 3<sup>rd</sup> party libraries.

### Webble Instantiation

When a Webble instance is created the first thing that happens is the initialization of the core element. That means creating the Webble Core Properties, which most is found in the Webble Directive and the Webble Core Controller which then will encapsulate the View for the specific Webble and make a call to the initiation function inside the controller for that Webble to create the customized parts that differs this Webble's template from all others.

In the controller.js you find the creation of all Webble slots. This is also the place where any customizations of menu targets are initiated and Interaction objects are set up, as well as any

code to control it all, including slot change reaction. The latter is managed via Webble World internal Event management or using AngularJS watches.

At this stage the Webble is all visible and ready to be interacted with.

## **Developing a Webble**

For your convenience there are template versions of all possible files you might want to use in the Webble Dev pack and they all contain one example of how they could be used.

Inside the code and scripts there are a lot of information, explanations, hints and suggestions on what to do when adding your own code.

There are many comments in the existing code to explain what each method is doing and why and quite often also how. So studying the code should be more than sufficient to get a grip on what's going on, and it is not only recommended, it is demanded of you, in order to be able to develop your own Webbles. Many comments and explanations can also be found in the core reference code and readme.txt, so feel free to roam around there as well when you feel the urge to do so.

So now you are ready to begin your development.

### **Creating a Webble**

1. Give the Webble Controller (in controllers.js) a suitable name and make sure that is also the name used in the view.html to connect them both.
2. Edit the view.html file to look like you want the finished Webble to look like, with your elements and content. Clean away unused stuff.
3. Edit the styles.css so that it contains the classes and styles you want to use inside your view.html
4. Edit the controller.js and bind values correctly, such as internal properties and slots to the content of the view.html. Configure CSS to slots sections to match your view. Also configure (or remove) the part for custom menu and interaction objects. Edit the initiation function and any other function your Webble will use or need to work properly.
5. If your Webble needs a custom directive, write that in directives.js and attach it to the view.html accordingly, either as a class, element or attribute.
6. If your Webble needs a custom filter, write that in filters.js and attach it to the view.html (and/or controllers.js) accordingly.
7. If your Webble needs a custom service, write that in services.js and call it from the controller, directive or filter accordingly.
8. If any other code file, library or content is required, link to it properly from where it is called for. 3<sup>rd</sup> party files should all reside together at the Webble root with no sub-folders.

9. If you developed in a local editor and are not already online, then login to your Webble World account online and upload all files needed to your sandbox as explained previously in chapter 2 (Figure 25).
10. Load and test your Webble. Both as stand-alone and in combination with others. Look for possible bugs.
11. If bugs or issues are found, go back to your code and edit where you believe it is needed.
12. Repeat 10 – 12 as much as you need until you think your Webble is bug free and amazing.
13. Select the Webble and pick ‘Publish’ from the menu as explained in chapter 2 (Figure 18), edit the data to be informative and helpful and submit.
14. Visit the Webble browser and rate your Webble as it is now online for all to see and share.

## Tips & Tricks & Basic Stuff

In this section we will present some tips and tricks that we have gathered while building our own Webbles. Solutions that maybe are not obvious at first but definitely do the job. As well as some basic stuff that one should really be aware of when developing Webble Templates.

### Event Listening

If you need to be informed when certain things happens to your Webble you can set up an event listener. There are two ways to do this. First is the internal Event Manager that listens to Webble specific events as shown below.

```
slotChanged  
deleted  
duplicated  
sharedModelDuplicated  
pasted  
gotChild  
peeled  
lostChild  
keyDown  
loadingWbl  
mainMenuExecuted  
wblMenuExecuted
```

You register your listener in the Webble using the following function.

```
$scope.registerWWEventListener(eventType, callbackFunc, targetId,  
targetData;
```

targetId can be undefined and it will then automatically listen to this Webble’s internal events, or one can set it to the instance Id of another Webble, most likely a child to catch events to that one, or set to null to listen for all Webbles having events and then sort out things on your own in the callback function.

targetData is only useful when listening to slot changes. One can then narrow down the firing of the event to happen only for a specified slot by adding the name of the slot as the targetData argument.

When the event fires it will call the registered callback function and return a data package with needed information of what has happened. The content of that package depends on the event.

```
slotChanged: {targetId: [Instance Id for webble getting slot changed],  
slotName: [Slot Name], slotValue: [Slot Value], timestamp: [a  
chronological timestamp value]}
```

```
deleted: {targetId: [Instance Id for webble being deleted],  
timestamp: [a chronical timestamp value]}
```

```
duplicated: {targetId: [Instance Id for webble being duplicated],  
copyId: [Instance Id for Webble that is a copy], timestamp: [a  
chronological timestamp value]}
```

```
sharedModelDuplicated: {targetId: [Instance Id for webble being  
duplicated], copyId: [Instance Id for Webble that is a copy],  
timestamp: [a chronological timestamp value]}
```

```
pasted: {targetId: [Instance Id for webble being pasted], parentId:  
[Instance Id for Webble that is pasted upon], timestamp: [a  
chronological timestamp value]}
```

```
gotChild: {targetId: [Instance Id for webble getting child], childId:  
[Instance Id for Webble that was pasted], timestamp: [a chronological  
timestamp value]}
```

```
peeled: {targetId: [Instance Id for Webble leaving parent], parentId:  
[Instance Id for Webble that lost its child], timestamp: [when it  
happened as ms integer]}
```

```
lostChild: {targetId: [Instance Id for Webble losing child], childId:  
[Instance Id for Webble that was peeled away], timestamp: [when it  
happened as ms integer]}
```

```
keyDown: {targetId: null[=UNSPECIFIED], key: {code: [key code], name:  
[key name], released: [True When Key Released], timestamp: [a  
chronological timestamp value]}
```

```
loadingWbl: {targetId: [Instance Id for webble that got loaded],  
timestamp: [a chronological timestamp value]}
```

```
mainMenuExecuted: {targetId: null[=UNSPECIFIED], menuId: [menu  
sublink id], timestamp: [a chronological timestamp value]}
```

```
wblMenuExecuted: {targetId: [Instance Id for the Webble executing  
menu], menuId: [menu item name], timestamp: [a chronological  
timestamp value]}
```

One can also use AngularJS \$watches, but that is mainly useful for internal value changes.

One important thing to be aware of is that `$watches` does not fire immediately when a value change but instead, when angular notices it has changed, which can be many cycles later and even after many intermediate value changes. When that is an issue than one should avoid using `$watches`. The internal event firing is instant and the user is called immediately. But in some cases using `$watch` is a better approach, for example when the Webble is called a little bit too early, before DOM changes has not yet occurred, then using `$watches` is more smooth. If you register a slot change listener before adding the slot you will be informed of the initiation value, but not if you register the listener after the add slot call. Also be aware that it is your job to make sure you do not create an infinite loop of catching triggers, react upon it and by that cause more events to trigger for eternity.

### **Custom Forms**

If you ever need to create and display any custom form window that can easily be done by using the same technique used with the already existing forms for properties, connections and so on. Just study the reference code and `readme.txt` a little bit and you should be on track within a few minutes. You can also study a sandbox copy of the Soap and Restful Service Client Webble and its 'Add Parameter' form. The principle is easy. You create the form in HTML and store it in the `view.html` file as a "text/ng-template" type. Create a controller for the form and store it in the `controller.js` file. And then add a trigger for the form to open which makes a call similar to this:

```
$scope.openForm('[FORM-NAME]', [{templateUrl: '[HTML-OBJECT-NAME]',  
controller: '[FORM-CONTROLLER-NAME]', size: ''}, {}], [CALLBACK-  
FUNCTION]);
```

### **Custom Messages**

Likewise, if you need to display some info, then you can use the same code the system uses by just making the proper call with the proper parameters. Once again, just study the reference code and `readme.txt`.

```
$scope.showQIM(qimText, qimTime, qimSize, qimPos, qimColor);
```

### **Cool custom JavaScript or AngularJS**

So you found this amazing library done by someone and you want to use it in your Webble. Well go ahead, do what you want, just remember to include the needed links and files where they are supposed to be in the `manifest.json` file and then it will all be fine and dandy.

### **Web Services, data bases and APIs'**

No problem, just include it as usual and call where you should, Webble loves getting connected to the outside world, as long as the outside world loves being connected to.

### **What is saved with the Webble?**

When a Webble is saved or published a JSON configuration file is created which contains most important parts of a Webble, but not everything are saved. For example internal non-slot values for a specific Webble are not stored, and if such exists and cannot be recreated in the Webble itself then the developer should store those values in the Webble definition file using



the `coreCall_CreateCustomWblDef` function in the Webble controller to add what is needed to be remembered by the Webble.

There are also a set of extra slot metadata, as for example the slots visibility status, which are not being stored either. The idea is instead that such metadata would be set at initiation of the Webble and would not be needed to save away, in order to keep the slot files as small as possible. For dynamic slots it is recommended to use an event listener to catch slots whose 'metadata' need to be set properly.

## **Summary**

I hope I have shown you all that you will ever need to be able to build your own Webbles now or at least pointed you in the right direction of where all you ever might need and want exist. If not, do not hesitate to contact us at [mkuwahara@meme.hokudai.ac.jp](mailto:mkuwahara@meme.hokudai.ac.jp) for questions, comments, requests, ideas or samples of your Webble work.

And with that I wish you the best of luck with your Webble development.

***Micke Kuwahara, Sapporo 2014***

# Index

## A

AngularJS.....9, 22, 57  
Architecture .....8, 58  
Assign/Revoke Parent .....20, 44

## B

Bring To Front..... 47  
Browse..... 21  
Bundle .....21, 33, 47

## C

Cancel.....12, 21, 28, 47  
Chat ..... 36  
Child .....20, 21, 45, 48  
Compound..... 9  
Connect Slots.....17, 21, 45  
Connection View .....21, 35  
Custom ..... 62  
Custom Slot ..... 49  
Custom Slots..... 21

## D

Delete ..... 21, 33, 44, 48, 54  
Deselect..... 32  
Desktop ..... 13, 26, 34, 35, 53, 54, 55  
Developers Pack ..... 22  
Documentation .....36, 54  
Download .....37, 55  
Duplicate ..... 15, 22, 24, 32, 44, 47, 48, 54

## E

Execution State ..... 22

## F

FAQ.....36, 37, 54

## G

Group ..... 48

## I

Instantiation ..... 58  
IntelligentBox .....7, 9  
IntelligentPad .....7, 8, 9, 25  
Interaction Balls ..... 43  
Interaction objects ..... 22

## L

Load.....22, 29  
Login.....22, 38, 39  
Logout .....43

## M

Main Menu.....23  
Meme Media.....1, 7, 8  
Merged Slot.....23  
Model ..... 24, 32, 47, 54, 58, 59  
Mouse .....55  
Multi-Webble Properties .....33

## P

Parent.....23, 44, 45, 48  
Popup menu .....44  
Properties.....13, 23, 31, 34, 45, 58  
Protection .....23, 47  
Publish.....23, 30, 44

## R

Recent .....30  
Redo .....32, 54  
Register .....22  
Report Bug .....38, 55  
Rescale .....44  
Rotate.....43

## S

Sandbox.....31  
Save .....44, 48  
Select.....12, 17, 32, 54  
Shared .....24, 32, 47, 48, 54  
Slots.....24  
Super parent .....24

## T

Template .....25  
Touch Screen.....55

## U

Unbundle .....47  
Undo.....31, 54  
Ungroup .....48  
Upload.....25, 30

<b>V</b>		Webble Id.....25
View.....	25, 54, 58	Webble Menu .....25
<b>W</b>		Webble Prototype.....26
Webble .....	25	Webble Template.....26
Webble Browser.....	11, 12, 21, 22, 28	Workspace .....26, 35

## Figure Index

FIGURE 1: OPEN THE WEBBLE BROWSER, WHICH CONTACTS THE WEBBLE WEB SERVICE.....	11
FIGURE 2: THE WEBBLE BROWSER IN ACTION .....	12
FIGURE 3: WE HAVE LOADED OUR FIRST WEBBLE.....	12
FIGURE 4: THE INFO-BUBBLE DISAPPEARS AFTER A FEW SECONDS. ....	13
FIGURE 5: YES, THERE IS A LOT YOU CAN DO, BUT STAY WITH PROPERTIES FOR NOW.....	13
FIGURE 6: YOU MAY CHANGE ANY PROPERTY YOU WANT, AND ALL CHANGES TAKE EFFECT WHEN THE FORM IS SUBMITTED.....	14
FIGURE 7: LOOKS A BIT NICER, DOESN'T IT? .....	14
FIGURE 8: DUPLICATION MAY ALSO BE DONE VIA THE WEBBLE'S CONTEXT MENU, BUT FOR NOW WE GO VIA THE MAIN MENU. ....	15
FIGURE 9: THREE INDEPENDENT WEBBLES READY FOR DUTY. ....	15
FIGURE 10: MAKING A CHILD - PARENT CONNECTION. ....	16
FIGURE 11: THE ARROWS INDICATE CHILD-TO-PARENT CONNECTIONS. A CHILD MAY ONLY HAVE ONE PARENT, BUT A PARENT MAY HAVE MANY CHILDREN.....	16
FIGURE 12: A WEBBLE NEEDS TO BE A CHILD IN ORDER TO HAVE THE 'CONNECT SLOTS' MENU ITEM AVAILABLE. ....	17
FIGURE 13: THE CHILD AND THE PARENT SLOT DO NOT NEED TO BE THE SAME BUT THEY COULD BE THAT TOO. ....	17
FIGURE 14: RESIZE WIDTH AND HEIGHT OF CHILD AND PARENT VIA SLOT CONNECTION. ....	18
FIGURE 15: SLOT COMMUNICATION IS A CORE PART OF THE ARCHITECTURE. ....	19
FIGURE 16: AS YOU SEE THERE ARE BASICALLY JUST YOUR IMAGINATION AND THE POWER OF THE SLOTS THAT LIMITS YOU.....	19
FIGURE 17: IF YOU LOOK FOR A WEBBLE THAT IS NOT THERE, MAYBE YOU CAN BUILD AND PUBLISH IT YOURSELF. ....	20
FIGURE 18: PUBLISHING A WEBBLE CAN ONLY BE DONE BY REGISTERED USERS. ....	24
FIGURE 19: DISPLAYED NAME CONSISTS OF THE USER CUSTOMIZED NAME, THE UNIQUE ID AND THE WEBBLE TYPE NAME. ....	25
FIGURE 20: AVAILABLE WORKSPACES ARE EITHER THOSE ONESELF SAVED PREVIOUSLY OR ONE THAT SOMEONE SHARED WITH THE USER.....	26
FIGURE 21: ANY NAME WORKS, AS LONG AS IT IS NOT USED BEFORE. THE SYSTEM WILL HELP YOU IF YOU ARE NOT SURE. ....	27
FIGURE 22: ONE DOES ALSO REMOVE ALREADY ADDED COLLABORATORS WITHIN THIS FORM BY SELECT USER AND CLICK REMOVE. ....	27
FIGURE 23: WHILE THE USER BASE IS STILL KIND OF SMALL, MANY WEBBLES MIGHT STILL BE UNRATED. MAYBE YOU CAN HELP OUT WITH THAT. ....	28
FIGURE 24: ANY WEBBLE ONLINE OR LOCALLY CAN BE LOADED THIS WAY. ....	29
FIGURE 25: IN THIS AREA YOU UPLOAD OR AUTO-GENERATE AND EDIT HTML5 FILES YOU HAVE CREATED TO BUILD A WEBBLE TEMPLATE AND ADDITIONAL INFORMATION ABOUT IT. IT WILL THEN APPEAR IN THE USERS SANDBOX OF HIDDEN WEBBLES, WHERE IT WILL STAY UNTIL PUBLISHED. ....	30
FIGURE 26: THE DEVELOPER CAN EDIT THE CODE DIRECTLY IN THE BROWSER AND IMMEDIATELY SEE THE RESULT AND POSSIBLE DEBUG INFORMATION WHEN LOADING THE WEBBLE INTO THE ENVIRONMENT.....	31
FIGURE 27: CHECK OR UNCHECK THE SLOTS THAT SHOULD BE SHARED BETWEEN SHARED MODEL DUPLICATES. ....	32
FIGURE 28: THE BUNDLE WEBBLE WILL ALSO HAVE SOME OF ITS OWN SLOTS FOR VISUAL CONFIGURATION OUTSIDE THE ONES THE USER PICK. ....	33
FIGURE 29: WEBBLE PROPERTY VALUE MANIPULATION FOR ONE OR MULTIPLE WEBBLES.....	34
FIGURE 30: FUTURE VERSIONS OF THE WEBBLE WORLD SYSTEM WILL PROBABLY PROVIDE A LARGER RANGE OF PROPERTIES.....	34

FIGURE 31: THE CONNECTION VIEW LAYER, WILL AUTOMATICALLY HIDE AGAIN WHEN THE USER STARTS INTERACTING WITH THE WEBBLES.....	35
FIGURE 32: SAVES YOU FROM COUNTING THE WEBBLES BY YOURSELF. ....	35
FIGURE 33: THE FAQ SECTION IS PROBABLY THE BEST AND PRIMARY ATTEMPT TO GET ANSWERS ON GENERAL QUESTIONS ABOUT WEBBLE WORLD.....	36
FIGURE 34: THE WEBBLE CHAT IS YET ANOTHER WAY OF GETTING IN TOUCH WITH OTHER WEBBLE WORLD USERS AROUND THE WORLD. ....	37
FIGURE 35: WHO MADE THIS EXTRAORDINARY PIECE OF SOFTWARE? .....	38
FIGURE 36: TYPE IN THE REQUESTED INFORMATION AND YOU ARE ONE STEP CLOSER TO PUBLISHING YOUR OWN WEBBLES. ....	39
FIGURE 37: AFTER LOGGED IN THE USER IS AUTOMATICALLY LOGGED IN AGAIN FROM THE SAME DEVICE AND BROWSER WHEN RETURNING TO WEBBLE WORLD. ....	39
FIGURE 38: CURRENTLY ALL THIS INFORMATION IS PRIVATE AND NOT DISPLAYED TO ANYONE ELSE, EXCEPT USERNAME. ....	40
FIGURE 39: IF YOUR USER IS ASSIGNED TO A GROUP A NOTIFICATION WILL APPEAR HERE.....	40
FIGURE 40: HERE YOU ASSIGN OR REVOKE YOUR TRUST TO GROUPS EXISTING IN WEBBLE WORLD AND THE WEBBLES THEY DEVELOP. ....	41
FIGURE 41: MORE ASSOCIATIVE SOCIAL MEDIA SITES MAY BECOME AVAILABLE IN THE FUTURE.....	41
FIGURE 42: MOST WEBBLES PROBABLY DO NOT REQUIRE THESE KEYS, BUT THOSE WHO DO, NOW HAVE A VERY GOOD WAY OF HOW TO DEAL WITH THAT. ....	42
FIGURE 43: GROUPS IN WEBBLE WORLD IS A WAY OF MAKING CONNECTIONS AND DEVELOP TRUST. ....	42
FIGURE 44: LANGUAGES AVAILABLE INCREASE BY TIME, SO IF YOUR LANGUAGE IS NOT PRESENT YET, PLEASE BE PATIENT. ....	43
FIGURE 45: TO DELETE A CUSTOM SLOT ONE JUST CLICKS THE X. ....	45
FIGURE 46: THE USER CHECKMARKS THE THINGS THIS WEBBLE SHOULD NOT BE ALLOWED TO DO. ....	47
FIGURE 47: ADDING CUSTOM SLOT CAN BE IMMENSELY HELPFUL WHEN BUILDING COMPLEX WEBBLES.....	49
FIGURE 48: JUST CLICK IN THE CROSS SECTION FOR THE INTERNAL SLOTS IN THE MERGED SLOT THAT YOU WISH TO SHARE VALUE. ....	50
FIGURE 49: AN EASY WAY TO GET THE WEBBLE TO DO A COMPLEX SET OF SLOT CHANGES WITH ONE CLICK.....	50
FIGURE 50: THE USER HAVE 12 INTERACTION OBJECTS AVAILABLE TO THEM TO MANIPULATE. ....	51
FIGURE 51: THIS INFORMATION IS MAINLY INTERESTING WHEN LOOKING TO IDENTIFYING THE AUTHOR OR ID OF A WEBBLE OR BUILDING A SIMILAR VERSION OF ITS TEMPLATE. ....	52

## Bibliography

Dawkins, Richard. *The Selfish Gene*. NY: Oxford University Press, 1976.

Tanaka, Yuzuru. *Meme Media and Meme Market Architectures: Knowledge Media for Editing, Distributing, and Managing Intellectual Resources*. NJ: IEEE Press & Wiley-Interscience, 2003.