Corso di Laurea Specialistica in Ingegneria Informatica

Tesi di Laurea

Development of an e-commerce platform and web 2.0 services for a high-tech university spin-off

Laureando

Francesco Pellegrini

Relatore

Prof. Federico Avanzini

Co-relatore

Günter Geiger

Anno Accademico 2009/2010

A mio nonno

You can know the name of a bird in all the languages of the world, but when you're finished, you'll know absolutely nothing whatever about the bird... So let's look at the bird and see what it's doing – that's what counts. I learned very early the difference between knowing the name of something and knowing something.

R. Feynman, educator & physicist

Contents

Sommario

Questa relazione di tirocinio tratta il lavoro svolto durante i sei mesi di internship presso **Reactable Systems**, a Barcellona, in Spagna.

Reactable Systems (vedi Capitolo 2) è un'impresa nata come spin-off dell'Università Pompeu Fabra di Barcellona. Fra le sue tante collaborazioni spicca quella con il Music Technology Group di Barcelona, uno dei maggiori laboratori di ricerca in tecnologia musicale.

Sebbene molto giovane, l'impresa può vantare diversi riconoscimenti mondiali ed alcuni prodotti rivoluzionari che hanno destato l'interesse di rinomati musicisti della scena internazionale, Björk tra tutti.

Appartenendo alla categoria *sound and music computing*, Reactable Systems potrà usufruire di nuovi canali comunicativi e verrà coinvolta nella nuova *knowledge economy*, in costante crescita ed evoluzione.

Durante il tirocinio sono stati sviluppati molti progetti, tutti aventi come *leitmotif* il **web development**.

I mesi iniziali sono stati dedicati allo sviluppo di un **forum** (vedi Capitolo 4), alla sua integrazione con il sito web dell'impresa, basato sul famoso Content Management System (CMS) WordPress, ed alla **creazione di plugin personalizzati** (vedi Sezione 4.8) per risolvere varie esigenze.

Una parte importante del tirocinio ha riguardato il **sito web ed il back-office** (vedi Capitolo 5), con la creazione di una sezione per la gestione degli ordini e delle prenotazioni e l'integrazione di un **software per newsletter** (vedi Sezione 5.5); inoltre alcune settimane sono state dedicate alla progettazione di una **nuova sezione del sito web** (vedi Sezione 5.3), destinata ad ospitare materiale riservato ai clienti registrati.

È stata effettuata una selezione di software open source per **piattaforme e-commerce** (vedi Capitolo 6), per poter successivamente sviluppare su una di esse.

Infine è stato realizzato un **tutorial per il Reactable** (vedi Capitolo 7) usando HTML5 e jQuery UI, con l'obiettivo di promuovere lo strumento ed avvicinare nuove persone al suo mondo.

Durante il tirocinio si è sempre lavorato seguendo i canoni dell'Ingegneria del Software [1] e sviluppando software open source nel rispetto degli standard WordPress per la codifica [2].

x

Chapter 1

Introduction

This stage report deals with the work carried out during the six months internship served at **Reactable Systems**, in Barcelona, Spain.

Reactable Systems (see Chapter 2) is a spin-off enterprise of the Pompeu Fabra University of Barcelona. Among its contributors, the Music Technology Group of Barcelona, one of the biggest research labs in music technology, is the most important.

Although very young, the enterprise has already received several worldwide awards for its revolutionary products, which have arisen the interest of several famous international musicians, Björk amongst all.

Being part of the sound and music computing field, Reactable Systems will be able to take advantage of new communication channels and will be involved in the new, constantly growing and evolving knowledge economy.

During the internship many projects have been carried out, all having **web development** as *leitmotif*.

The first months were dedicated to the development of a **forum** (see Chapter 4) that had to be integrated with Reactable Systems website, built on the famous Content Management System (CMS) WordPress, and to the **creation of custom plugins** (see Section 4.8) to fulfill several needs.

An important part of the stage centered on **the website and the back-office** (see Chapter 5): a section for orders and reservations management was realized and a **mailing list software** (see Section 5.5) was seamlessly integrated in the resident architecture; also, some weeks were spent on the design a **new section of the website** (see Section 5.3), which was going to house content reserved to registered customers.

An open source **e-commerce platforms** software selection has been carried out (see Chapter 6), to subsequently begin developing on one of them.

Last, a **tutorial for the Reactable** (see Chapter 7) was created using HTML5 and jQuery UI, to promote the instrument and introduce new people

to its world.

During the whole internship, Software Engineering standards [1] were the rule, while open source software was developed following Wordpress coding standards [2].

Chapter 2

The enterprise

2.1   Reactable Systems

Reactable Systems is a spin-off enterprise of the Pompeu Fabra University. It was founded by Sergi Jordà, Marcos Alonso, Günter Geiger and Martin Kaltenbrunner and it is collaborating with the Music Technology Group of Barcelona, one of the biggest research labs in music technology.

Its first and most notable product was the **Reactable** (see Section 2.2), a novel musical instrument, conceived and developed since 2003, that has gained international reputation and received several **worldwide awards** like the Golden Nica, two D&AD Yellow Pencil, Rolling Stones' "Hot Instrument of the Year" 2007 and many more, becoming one of the most popular and worldwide acclaimed new musical instruments of the early 21st century.

Several famous international musicians have expressed their interest in the enterprise, Björk amongst all.

2.2   The Reactable

The Reactable [3] is an innovative instrument, designed to be **collaborative, intuitive and musically challenging**: its technology is completely transparent to the user, since no mouse, keyboard, cables nor wearables are featured.

The Reactable architecture (see Figure 2.1) revolves around a translucent round table: beneath it, an infra-red camera constantly analyses the surface, tracking hand gestures and tangible objects tagged with *fiducials*; collected data is processed and then transmitted to an audio synthesizer,
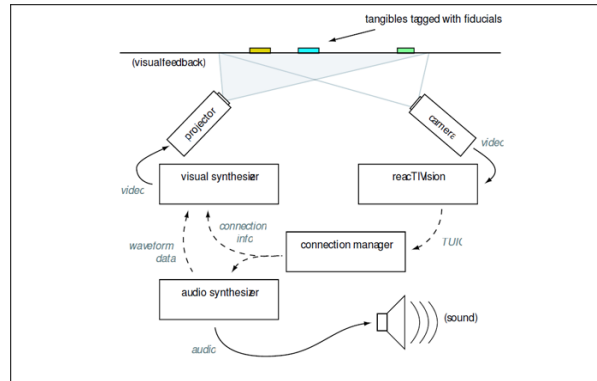
Figure 2.1: The Reactable architecture

while a projector draws dynamic animations, providing visual feedback.

The way the Reactable works is deeply inspired by **modular analogue synthesizers** such as the early 60s Robert Moog's: the Reactable enhances them with new concepts of sampling and digital effects processing.

While with modular synthesizers you had to connect these different modules with patch cables in a complex and error-prone process, with the Reactable this process is done in a **much easier and more intuitive** way, since connections between tangibles are managed automatically depending on their type, affinities and proximity. Each tangible has a certain number of different in-output connectors and periodically checks its neighborhood for compatible tangibles having available ports. Thus, moving a tangible around the table surface alters existing connections, resulting in an **highly dynamic environment**.

When placed on the surface, a tangible illuminates and starts to interact with neighboring tangibles, according to their positions and proximity. These interactions are visible on the table surface, whom acts as a screen, giving instant feedback about what is currently going on, **turning music into something visible and tangible**.

The Reactable also allows free finger-drawing on the table surface, making defining envelopes, wave-tables or spectra an extremely easy and intuitive task. All of the tasks can be performed by **any number of players**, with no need to previously define user roles.

Speaking of audio synthesis, the Reactable features high level objects, such as PureData ones [4], and lower level sound synthesis and processing algorithms and techniques. Reactable objects were initially grouped into six categories: Generators, Audio Filters, Controllers, Control Filters, Mixers and Clock Synchronizers [5].

The visual feedback is handled by the reacTIVision, an open source soft-

ware developed by Ross Bencina of the Music Technology Group; it provides an high temporal resolution for the tracking of fast movements on the Reactable surface. Collected data is sent to client application via TUIO, a protocol based on OpenSound control; it includes position, absolute angle of each tangible and also distance and angle relative to its closest neighbors.

See also [6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19].

2.2.1 The Reactable Experience

The Reactable Experience (see Figure 2.2) is a version of the Reactable specially designed for public spaces and institutions such as museums, science centers, schools and universities.

Conceived from the original Reactable, which was designed for professional musicians, the Reactable Experience is an **intuitive and robust version for casual players and children**. Its design is so intuitive that can be enjoyed from the very first minute by casual players, but at the same time its richness allows never ending creative possibilities for experienced musicians.

The Reactable main characteristic is its tangible interface: the instrument is based on a translucent and luminous round table. Putting tangibles on the surface, turning them and connecting them to each other, performers can combine different elements like synthesizers, effects, sample loops and control elements creating a unique and flexible composition.

The Reactable Experience is already in use in several art galleries and science centers around the world, such as Unites States of America, Canada, United Kingdom and Germany, where it is always one of the main attractions.

Figure 2.2: The Reactable Experience

2.2.2   The Reactable Live!

The Reactable Live! (see Figure 2.3) is the new incarnation of the Reactable, the result of many years of research in computer music and man-computer interaction. Officially released on July, 2010, it is specifically **designed for musicians and live performers**. Aside from the Reactable Experience, it can be bought online on Reactable Systems website.

The Reactable Live! interface is clean and direct, without text lists or endless configuration menus: it puts the whole music creation process in the musician hands. Its being multi-touch makes it collaborative and intuitive. This portable version is nevertheless robust, flexible and easy to set up.

Figure 2.3: The Reactable Live!

2.2.3   The Reactable mobile

On October 2nd, 2010[1], Reactable Systems released a version of its software for Apple mobile devices (iPod touch, iPhone and iPad), named Reactable mobile (see Figure 2.4). Taking full advantage of the multi-touch technology and featuring the very same audio and graphics engine of the original Reactable, this application **brings a full creative experience onto mobile devices**.

In the beginning, tangible were grouped into four different categories: Sound Generators (e.g., Oscillator), Effects and Filters (e.g., Delay), Controllers (e.g., Sequencer) and Global Objects (e.g., Tonality). Besides, the Reactable mobile also features an **accelerometer** and a **microphone input**, widening the yet very intuitive and interactive multimedia user experience.

---

[1]http://www.reactable.com/introducing-reactable-mobile/

Figure 2.4: The Reactable mobile

Moreover, it allows to import loops and samples, define new sequences and store them on the device; sets can then be exported and shared with other users.

2.2.4 Upcoming products

A brand new product, named **Reactable Mediabench** (see Figure 2.5), is also under development: it is described as "the perfect tool for presenting content using multi-touch technology". Watching videos, displaying slide-shows, browsing Google Maps, Google Earth or any other web content will never be the same.
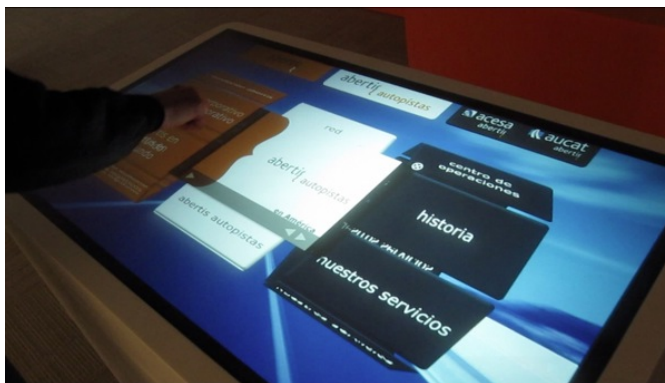
Figure 2.5: The Reactable Mediabench

2.3   Industrial context

Sound and music computing has always been an important research field
for the music related industry, driven by the growth of the Information and
Communication Technologies (ICT) sector and the innovations coming out
of it. These industries, in the middle of important changes, are trying to
adapt to the new markets and exploring the potential of these technologies.

Traditionally there have been two distribution channels for media content,
namely, physical distribution and analogue broadcasting (radio, tv). Now,
thanks to the new ICT technologies, we may take advantage of **new chan-
nels** such as Internet, mobile communications, digital TV and radio [20].
Both the production and consumption of music related goods is now global-
ized and international cooperation is more important than ever [21].

Music related activities are part of the **new knowledge economy** and
they should take advantage of the continuing growth of this sector, though
new models of music content protection are also needed, to control the use
of intellectual property rights [22].

The ICT technologies open to new, yet unexplored, possibilities, com-
pletely reshaping the music business and the sound and music computing
field will play a **crucial role in these developments** [23].

Chapter 3

Development environment

3.1 Apple Macintosh

All applications were developed and managed on an Apple Mac Mini desktop computer. Macintosh personal computers come with many pre-installed and built-in software dedicated to audio, video and software developers.

During the internship, the following software have been used.

3.1.1 Apache HTTP server

Apache HTTP Server (commonly Apache) is web server software notable for playing a key role in the initial growth of the World Wide Web. Apache is primarily used to serve both static content and dynamic Web pages on the World Wide Web, maintaining performances comparable to other high-performance web servers.

Many web applications are designed expecting the environment and features that Apache provides. Apache supports a variety of features, many implemented as compiled modules which extend the core functionality. Some common language interfaces support Perl, Python, Tcl, and PHP.

During the internship, Apache 2.2 was used.

3.1.2 MySQL

MySQL is a Relational Database Management System (RDBMS) that runs as a server providing multi-user access to a number of databases. MySQL works on many different system platforms, and all major programming languages with language-specific APIs include libraries for accessing MySQL databases.

Free-software projects that require a full-featured database management system often use MySQL. Its popularity for use with web applications is closely tied to the popularity of PHP, which is often combined with MySQL.

During the internship, MySQL 5.0 was used.

3.1.3   PHP

PHP (Hypertext Preprocessor) is a widely used, general-purpose scripting language that was originally designed for web development to produce dynamic web pages. For this purpose, PHP code is embedded into the HTML source document and interpreted by a web server with a PHP processor module, which generates the web page document.

As a general-purpose programming language, PHP code is processed by an interpreter application in command-line mode performing desired operating system operations and producing program output on its standard output channel. PHP is available as a processor for most modern web servers and as standalone interpreter on most operating systems and computing platforms.

During the internship, PHP 5.2 has been used.

3.2   phpMyAdmin

phpMyAdmin is a open source software tool written in PHP intended to handle the administration of MySQL over the World Wide Web. It supports a wide range of operations with MySQL like creating, modifying or deleting databases, tables, fields or rows.

phpMyAdmin has won several awards, it was chosen as the best PHP application in various awards and every year wins the SourceForge.net Community Choice Awards as "Best Tool or Utility for SysAdmins".

During the internship, phpMyAdmin 3.1 and 3.2 have been used.

3.3   WordPress

WordPress is an open source blog publishing application powered by PHP and MySQL which can also be used for content management. Used by over

2% of the 10.000 biggest websites, it is the most popular blog software in use today.

WordPress started as just a blogging systems, but has evolved to be used as full Content Management System (CMS) and so much more through the thousands of plugins, widgets, and themes available.

During the internship, WordPress 2.9 and 3.0 have been used.

3.4 Other software

3.4.1 Browsers

Websites layouts get very often displayed on screen in different ways depending on client browser, because browsers architecture may differ. This is why web applications are usually **tested on more than one browser**.

During the internship, the following web browsers have been used:

- Mozilla Firefox 3.6

- Apple Safari 4.0

- Google Chrome 5.0

- Microsoft Internet Explorer 7.0

3.4.2 Editors

Source code was modified with both TextWrangler and Komodo Edit, which feature multi-language file support, syntax coloring, multi-document editing, extendibility via plugins and much more.

One of TextWrangler's strengths is its flexible `grep` style pattern-based search and replace based on Perl-Compatible Regular Expression (PCRE), while Komodo Edit's auto-complete and call-tips functions make it one of the best allies of any developer.

During the internship, TextWrangler 3.1 and Komodo Edit 5.2 have been used.

LATEX was written and compiled with TEXworks, a powerful free editor included in the MacTEX distribution.

TEXworks includes an integrated PDF viewer which supports source/preview synchronization (i.e., control-click within the source text to locate the corresponding position in the output preview, and vice versa).

During the internship, TEXworks 0.2 and 0.3 have been used.

Chapter 4

Forum

4.1 Requirements

Reactable Systems needed a forum to ease customers feedback and overall communications.

The forum software had to:

- be open source;

- be easy to use;

- be flexible;

- feature the ability to restrict content;

- fit the needs of a Small and Medium Enterprise (SME), i.e., not to be over-sized;

- integrate seamlessly with WordPress;

- keep its users database separate from WordPress'.

4.2 Software selection

A web-wide **software selection** was performed, discarding *a priori* charged software, like vBullettin or IPBoard, though consciously knowing these usually come with better documentation and customer support.

4.2.1   bbPress

bbPress (see Figure 4.1) is an open source forum software designed to be fast, easy to use and highly customizable. Created by WordPress development team, it can be integrated with it, sharing its users database and functions.
bbPress 1.0 and 1.1-alpha (trunk) have been installed and tested.
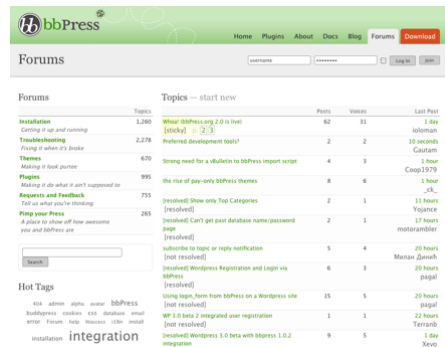
Figure 4.1: bbPress

bbPress' WordPress-like structure and functions was very easy to get familiar with. Even if it is a rather young project, bbPress features a wide online plugins repository and an excellent support forum which both were a great help during local tests.

4.2.2   phpBB

phpBB (see Figure 4.2) is a popular Internet forum package written in PHP, featuring modular design, high security, multiple-language interface, support for a multitude of database servers and complete layout customization, all with a low execution overhead.

Figure 4.2: phpBB

phpBB 3.0 has been installed and tested.

phpBB is usually recommended for large communities, and its features resulted too much for Reactable Systems' needs. Moreover, there were some errors during and after local installation, which made phpBB last version look not yet ready for a wide web launch.

4.2.3 Simple:Press

Simple:Press (see Figure 4.3) is a feature rich forum plugin for WordPress. It fully integrates into WordPress based sites utilizing the same user records and database and displaying on a single page. It is fully customizable and scalable, and also comes with a number of skins and sets of icons.

Simple:Press 4.2 has been installed and tested.

Figure 4.3: Simple:Press

Simple:Press was discarded because it shared WordPress users database, a feature Reactable forum had not to have for security reasons.

4.2.4 Lussmo Vanilla

Vanilla (see Figure 4.4) is an open source, standards-compliant, multi-lingual, theme-able, pluggable discussion forum by Lussmo. Though it is very easy to use even for beginner users, skilled developers will be able to highly customize Vanilla behavior and look thanks to its excellent configuration files.

Vanilla 1.1 has been installed and tested.

Vanilla looked way too plain and, according to online reviews, its plugins tended to crash the forum after every software update.

On July 22nd, 2010[1], Vanilla 2 was released, unfortunately too late for the evaluation process.

---

[1]http://vanillaforums.org/blog/vanilla-2-released/

Figure 4.4: Lussmo Vanilla

4.3   Development

bbPress was chosen as Reactable System forum software after a long pe-
riod of closed beta testing. Its simple structure, WordPress-like functions,
extensibility via plugins and excellent online community made it stand.

In the beginning, bbPress was installed in a sandbox (i.e., a private de-
velopment environment where coders may test software and scripts without
having to worry about their consequences) and later exported to a private
section of the online server, for further testing.

To speed up the release of its new version, **support was provided** at
bbPress official Trac repository: several tickets were created, closed or help
was supported.

WordPress coding standards [2] were the rule during the whole develop-
ment process.

4.4   Layout

Cascading Style Sheets (CSS) files were first in line, since the forum had to
look very close to Reactable Systems website: resident WordPress CSS was
therefore mirrored into bbPress one.

Luckily, many CSS tags could be copied without changes, but there were
many others that had to be modified by hand, checking step by step their
effects on the forum layout.

Subsequently, several changes took place to fulfill project needs till the
forum got its final look (see Figure 4.5).
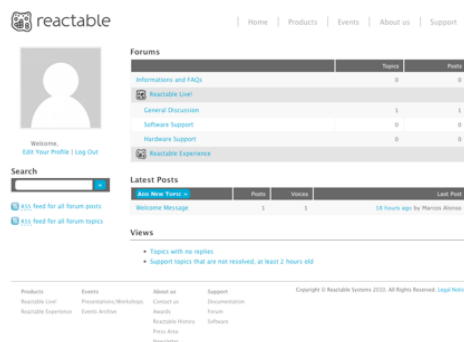
Figure 4.5: Reactable support forum

4.5 Structure

Reactable Systems wanted the forum to be customer-friendly, to have all its content categorized and its pages under moderators control. It had to have dedicated areas for each of the enterprise's main products and sub-forums for FAQs and other generic informations.

4.5.1 Content categorization

bbPress allows administrators to define a forum as a **Category**: Categories are forums where new topics cannot be created. Categories usually contain a group of sub-forums.

A forum for generic informations plus a Category for each Reactable Systems product were therefore created, each of which contained few sub-forums.

4.5.2 Hierarchy

bbPress features an option named **pretty permalinks** which makes links more human-friendly. However, this feature does not work exactly as one would expect to: if bbPress is located at `forums/`, one of its topics can be found at `forums/topic/topic-name`, while a forum link will look like `forums/forum/forum-name/`.

Thanks to **Nicer Permalinks** custom plugin (see Sub-section 4.8.3), forums links could be simplified and topics could more correctly look as contained in their parent forum.

4.6   Customers support

Reactable Systems wanted some forums to be used for customers feedback, thus these topics would have to act as question-answer threads.

Support forum plugin by A. Naik and S. Bauers realized exactly this wish. It turns forums into support forums and gives users and moderators the ability to mark topics as resolved, not resolved or not a support question. It helps customers to make questions and moderators to locate them faster and answer them precisely.

Unfortunately, this plugin was designed for bbPress 0.9, and referred to old data and deprecated functions. It also make use of a lot of database queries to retrieve settings and informations. After some failed attempts to get in touch with its developers, it was decided to modify the plugin and publish a faster, bbPress 1.0 compatible version named **Support Forums** (see Sub-section 4.8.4).

4.7   Privacy protection

By default, bbPress does not feature the option to hide topics or forums, but a couple of plugins that work exactly like this were found at bbPress official Plugin Browser.

A. Naik's Private Forums plugin hides private forums from users not matching appropriate role. Forum administrators can set the role to which forums are restricted to in the plugin configuration page. Unfortunately this plugin was not compatible with bbPress latest release.

Another plugin with the same aim was Hidden Forums by _ck_: it turns forums completely hidden except for selected members or roles. It was updated last on May 13th, 2009, with a note stating it was now bbPress 1.0 compatible. Unfortunately it did not feature a back-office panel, so forum administrators had to manually edit plugin source to add or remove forums to the hidden list.

A more complete solution was finally found in F. Bigiarini's **zaerl Visibility** plugin, which lets administrators hide or lock forums, topics and even user profiles to users or entire roles.

bbPress by default does not feature the option to define new user roles beside the default ones. This problem was quickly solved thanks to **Role Manager** plugin by Ben L., which grants administrators the ability to edit roles capabilities and to define new user roles.

Overall security was then enhanced via two custom plugins: **Swirl Unknowns** and **Deny Pages** (see Sub-section 4.8.5 and Sub-section 4.8.2),

which, respectively, turn the whole forum private and restrict access to pages and/or URLs.

4.8 Custom plugins

One of bbPress most powerful features is its **extendibility**: users can create plugins, share and improve them with the help of the online community. bbPress website features a dedicated section where anyone can download hundreds of free open source GPL-licensed plugins.

Several of these have been downloaded and integrated in the forum, but some needs could not be fulfilled by existing plugins, so it was decided to spend some weeks studying online documentation and analyzing existing source code to **develop custom plugins**.

4.8.1 Custom Usermeta

Custom Usermeta plugin (see Listing 4.1) was inspired by F. Bigiarini's zaerl Simple Registration plugin. It lets forum administrators add or remove meta from user profile all across the forum, easing registrations, user details editing and displaying.

```php
function custom_usermeta( $fields, $context ) {
  unset( $fields['occ'] );
  unset( $fields['interest'] );

  $fields['company'] = array(
    false,
    __( 'Company' ),
    'text'
  );

  $fields['reactable_model'] = array(
    false,
    __( 'Reactable model' ),
    'text'
  );

  return $fields;
}
```

Listing 4.1: custom-usermeta.php (fragment)

4.8.2 Deny Pages

Deny Pages plugin (see Listing 4.2) was created to extend the features of Disable Registrations plugin by S. Wheatley, that is to enhance forum security blocking every user attempt to get to restricted pages and/or URLs. These can be defined in the plugin configuration page by forum administrators.

Thanks to this plugin, any attempt to access bbPress registration module or other undesired pages was prevented.

```php
function deny_pages() {
  $denied = bb_get_option( 'deny_pages_settings' );

  if ( !is_array( $denied ) )
    return;

  if ( array_key_exists( 'pages', $denied )
    && !empty( $denied['pages'] ) ) {
    $denied_pages = $denied['pages'];

    $denied_pages = str_replace( '-', '\-', $denied_pages );
    $denied_pages = str_replace( '.', '\.', $denied_pages );
    $denied_pages = str_replace( '/', '\/', $denied_pages );

    if ( preg_match( '/('. $denied_pages .')\.php/i',
      $_SERVER['REQUEST_URI'] ) ) {
      global $bb;

      bb_safe_redirect( $bb->uri );
    }
  }

  if ( array_key_exists( 'urls', $denied )
    && !empty( $denied['urls'] ) ) {
    $denied_urls = $denied['urls'];

    $denied_urls = str_replace( '-', '\-', $denied_urls );
    $denied_urls = str_replace( '.', '\.', $denied_urls );
    $denied_urls = str_replace( '/', '\/', $denied_urls );

    if ( preg_match( '/('. $denied_urls .')/i',
      $_SERVER['REQUEST_URI'] ) ) {
      global $bb;

      bb_safe_redirect( $bb->uri );
    }
  }
}
```

Listing 4.2: deny-pages.php (fragment)

4.8.3 Nicer Permalinks

Nicer Permalinks plugin (see Listing 4.3) extends and enhances A. Mohta and M. R. Henderson's excellent plugin, named Remove Forum Topic, which rewrites forum URIs and emphasizes hierarchy.

Using Apache's `mod_rewrite` module and adding custom filters to bb-Press, forums URIs become `bbPress-path/forum-name/` while its topics are referenced by `bbPress-path/parent-forum-name/topic-name`.

Nicer Permalinks backups resident `.htaccess` when it gets enabled and also restores it automatically when gets deactivated, thanks to a bbPress hook which plugins can grab and bind to a function.

Since version 4.0, Nicer Permalinks features a configuration page where forum administrator may check if all prerequisites are met before enabling the plugin.

In version 5.0, all of the plugin filters were revised, **its performance greatly enhanced** and its installation made easier: thanks to bbPress global variables, data could be retrieved faster, with no need of making queries to the database.

Since April 16th, 2010, Nicer Permalinks is **available for download** at bbPress official Plugin Browser.

```php
function get_topic_nicer_link( $link ) {
  global $topic;

  if ( is_string( $topic ) || !$topic ) {
    $topic = bb_get_topic_from_uri( $link );

    if ( !$topic )
      return add_query_arg( 'view', 'all', wp_get_referer() );
  }

  global $bb;

  return str_replace( $bb->uri . 'topic', $bb->uri .
    get_forum( $topic->forum_id )->forum_slug, $link );
}
```

Listing 4.3: nicer-filters.php (fragment)

4.8.4 Support Forums

Support Forums plugin (see Listing 4.4) turns any number of forums into support forums, where users can mark topics as *resolved, not resolved* or *not a support question.*

It is possible to add icons to support topics, as well as to closed and sticky ones. A view for each support status may also be generated.

The plugin is based on so1o, mdawaffe and S. Bauers' Support Forum plugin, which has not been updated for bbPress 1.0, used deprecated data and functions and had performance issues.

This re-edition of the plugin features a revised configuration page, the ability to easily define and use custom icons, **drastically improved performances**, custom filters for more extensibility and is fully compatible with bbPress 1.0.

Since version 0.2, **backwards compatibility** was granted by a legacy importer whom retrieves Support Forum plugin settings.

Since July 16th, 2010, Support Forums is **available for download** at bbPress official Plugin Browser.

```php
function setTopicStatus( $topic_id = 0, $status ) {
  if ( false === (bool) $this->isEnabled() )
    return false;

  if ( !$topic_id )
    return false;

  $status = apply_filters( 'support_forums_set_topic_status',
    $status, $topic_id );

  if ( !array_key_exists( $status, $this->statuses ) )
    return false;

  return bb_update_topicmeta( $topic_id,
    'topic_support_status', $status );
}
```

Listing 4.4: support-forums.php (fragment)

4.8.5 Swirl Unknowns

Swirl Unknowns plugin (see Listing 4.5) was the first custom plugin developed for Reactable Systems. Based on M. D. Adams' Force Login plugin, it redirects every non-logged-in user to a custom page.

Swirl Unknowns features a configuration page where forum administrators can define the swirl page. The plugin checks if user is requesting an allowed page: if not, it redirects him to the swirl page instead, using bbPress' `bb_safe_redirect()` function.

Since version 0.7, Swirl Unknowns features a list of percent-substitution tags which ease defining, storing and processing the swirl page.

Reactable Systems version also featured a custom line of code simulating bbPress' login form standard redirection.

Since April 8th, 2010, Swirl Unknowns is **available for download** at bbPress official Plugin Browser.

```php
function swirl_unknowns () {
  global $swirl_unknowns_settings;

  if ( false === (bool) $swirl_unknowns_settings->isEnabled () )
    return;

  $swirl_page = $swirl_unknowns_settings->swirl_page;
  $immunes =      $swirl_unknowns_settings->immunes;

  foreach ( $immunes as &$immune ) {
    $immune = str_replace( '-', '\-', $immune );
    $immune = str_replace( '.', '\.', $immune );
    $immune = str_replace( '/', '\/', $immune );
  }

  $pattern = implode( '|', $immunes );

  foreach ( $swirl_unknowns_settings->tags as $tag => $value )
    $swirl_page = str_replace( $tag, $value, $swirl_page );

  if (
    !bb_is_user_logged_in ()
    && !preg_match( '/('. $pattern .')/i',
      $_SERVER['REQUEST_URI'] )
    && strcasecmp( $_SERVER['REQUEST_URI'], $swirl_page )
  ) {
    $swirl_page = add_query_arg( 'redirect_to',
      $_SERVER['REQUEST_URI'], $swirl_page ); // Reactable only
    wp_redirect( $swirl_page );
    exit;
  }
}
```

Listing 4.5: swirl-unknowns.php (fragment)

Chapter 5

Website and back-office

5.1 Architecture

Reactable Systems website is based on the famous Content Management System (CMS) **WordPress** (see Section 3.3): it stores pages content in a MySQL database, making website layout easy to change with no need to edit every page by hand.

WordPress pages are written in PHP, a programming language processed server-side that allows programmers to save code and time, and pages to be dynamic and user-transparent. It is also possible to **extend** WordPress functionalities via plugins or to edit its pages manually, since it is an open source software.

Reactable Systems server also included a private custom **back-office** section, where administrators could access reserved informations, like customers data and website statistics.

As a side note, **neither PHP nor WordPress had been used before**, both have been studied and experienced during the internship.

5.2 Software update and code cleansing

On June 17th, 2010[1], the thirteenth major version of WordPress, "Thelonious", was officially released by AutoMattic. It featured a new lighter interface, contextual help on every screen, and more than a thousand bug fixes and feature enhancements.

The decision of upgrading Reactable Systems website was hard to take, because the enterprise feared custom theme or embedded data could not work

---

[1]http://wordpress.org/news/2010/06/thelonious/

on the new version, but it was finally taken on July 16th, 2010.

Reactable theme was **cleaned up and upgraded** to make use of the new template functions added along with WordPress' new default theme, "Twenty Ten". An overall pages cleansing and revision was also performed. Apparently no change could be noticed, but overall loading time, cleanness of code and forward compatibility took **great advantages** by this software update.

5.3   Support section

Besides the forum (see Chapter 4), Reactable Systems needed other resources, like software updates and products manuals, to be **reserved** to registered customers. A new website section, where pages were listed but content could only be accessed after being logged in.

To check users credentials, every support page had **bbPress' login form** embedded in it (see Figure 5.1), since customers were registered in the forum database. To call bbPress' `bb_is_user_logged_in()` function, one of bb-Press configuration files had to be included into support pages source code (see Listing 5.1), increasing its overall loading time by a little.

This could be avoided if permissions were checked against WordPress users database, but this was **expressly forbidden** by the enterprise, because would have led to security issues on the main website.

```php
// Support page requested
<?php require_once ABSPATH . 'forum/bb-load.php'; ?>
<?php if ( bb_is_user_logged_in() ) : ?>
  // Display page content
<?php else : ?>
  // Display page excerpt
  <?php login_form(); ?>
  <link rel="stylesheet"
    href="<?php bb_stylesheet_uri(); ?>" />
<?php endif; ?>
```

Listing 5.1: page.php (fragment)

Support section was put online on July 26th, 2010, though no Reactable Live! customers had the required credentials to see its contents yet. A **special registration page** (protected by another login form) was made available, where customers could create their personal user profile for Reactable Forum and thus access the support pages.
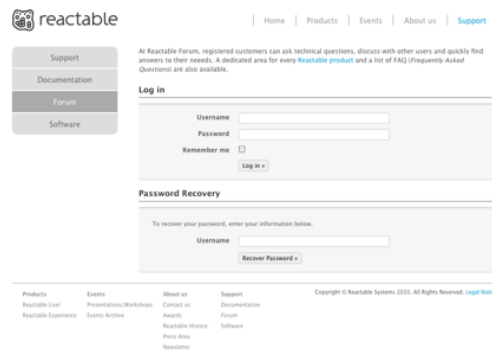
Figure 5.1: Support section login form

5.4 Pre-orders

Pre-orders for the Reactable Live! were going to open on March 29th, 2010, so order form and database structure were to be designed and tested very quickly. Only twenty units were going to be sold and only to customers from European Union. Other reservations were queued in a waiting list for future releases.

5.4.1 Back-end

Two SQL tables were created (see Figure 5.2): the active orders one an the customers one, listing everyone interested in the product purchase.

The new tables were added to resident MySQL database and a dedicated administration page was **seamlessly integrated** into Reactable Systems back-office. An SQL table, named `order_statuses`, was created to ease processing status changes: it associates every order status to its following in the processing flow. This way, changing an order status required no knowledge of exact statuses sequence.

The new back-office section consisted in 4 tables:

1. **Orders**, listing active orders;

2. **Customers**, listing all registered users;

3. **Reservations queue**, listing non-E.U. orders and late orders (i.e., having `order_id` greater than twenty);

4. **Orders archive**, listing non-paid and completed orders.

Table	Button	Action
Orders	+	Displays detailed customer and related orders informations. Some fields can be modified.
Orders	Details	Displays detailed order informations. Some fields can be modified.
Orders	Process	Sets order status to next one, according to MySQL's `order_statuses` table.
Orders	×	Sets order status to *not paid*.
Customers	Details	Displays detailed customer and related orders informations. Some fields can be modified.
Customers	Reserve	Sets customer status to *reserved*.
Reservations queue	+	Displays detailed customer and related orders informations. Some fields can be modified.
Reservations queue	Process	Creates a new order with *payment pending* status for selected customer and also sets customer status to *reserved*.
Reservations queue	×	Sets customer status to *not reserved*.
Orders archive	+	Displays detailed customer and related orders informations. Some fields can be modified.
Orders archive	Details	Displays detailed order informations. Some fields can be modified.

Table 5.1: Back-office functions

Every table entry showed just a little informations, though complete details and custom notes could be displayed and edited in a dedicated page, which also included customer's orders (see Table 5.1).

A function to change the status of more than one order at time was also implemented, but it was then discarded since not really needed.

A small jQuery script to control some user actions and reduce hard-coded data was also added.
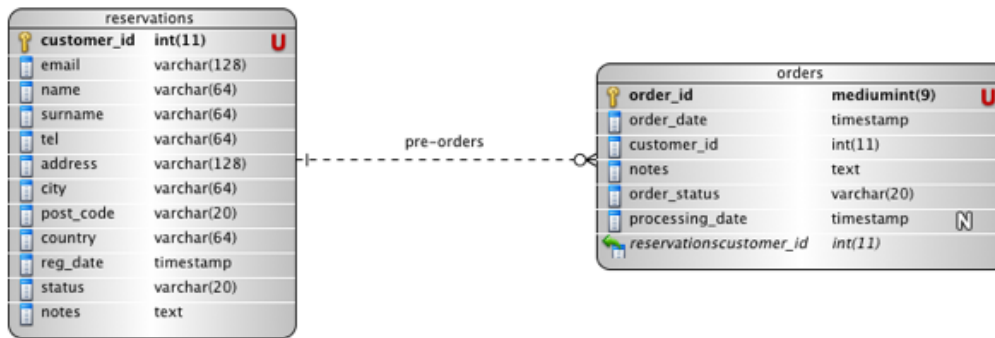
Figure 5.2: Pre-orders entity-relationship diagram

5.4.2 Front-end

Once back-end was done, registration form was next in line. A very important feature it had to have was to handle many reservations at time: never had two customers to get the same id, to preserve **database consistency**.

Front-end form was thus split in two forms, the former requesting just email, home country and *captcha* test, the latter containing the other fields. When a customer submits the first form, a new database entry is immediately inserted and he may fill the second one with no hurry. Forms keep track of customer id saving it for the current session thanks to PHP's *superglobal* array `$_SESSION[]`.

In the end, it was decided not to send automatic confirmation emails to registered customers, but to manage it manually instead, making customer relations **more personal**.

On April 27th, 2010, the pre-order form went under **restructuring** (see Figure 5.3): the two previous forms were merged into one, easing the regis-

Figure 5.3: Reservations form for the Reactable Live!

tration process and signaling pre-orders were closed and, from that moment on, only reservations were allowed.

5.5   Mailing list software

Reactable Systems needed a software to send newsletters and promote its brand new product, the Reactable Live! (see Section 2.2.2).

The mailing list software had to:

- be open source;

- schedule mailings, not being marked as spam by servers;

- feature the ability to import contacts;

- easily integrate with resident back-office structure;

- send mailings both in HTML format and plain text.

5.5.1   Simple Mailing List

After a brief software research, NotOneBit.com's **Simple Mailing List** (SML) was found: it is an open source mailing list manager written in PHP, designed to be both simple and powerful. It features mailing scheduling, sending mailings in HTML format and plain text, easy contacts management and mailings preview.

Simple Mailing List online demo was tested for a few days and then version 1.5 was downloaded and installed: thanks to its clear tabbed interface, it was easily and **seamlessly integrated** into Reactable Systems back-office architecture (see Figure 5.4) and its well commented source code provided help during configuration, customization and integration with the database.

Simple Mailing List was configured to use Reactable Systems customers database and then a `.csv` (Comma-Separated Values) file consisting of more than eight thousands email addresses was imported. To fit the new database structure, the file had to go through **many processing steps**: first, the `.csv` was converted into Macintosh Address Book format using an import/export utility named Abee, then it was exported in vCard format. Unfortunately, the export function added many lines of unnecessary code that had to be removed. Thanks to TextWrangler's (see Sub-section 3.4.2) regular expression search and replacement, this was done pretty quickly. A new field named *source* was also added to all email addresses. Finally the list was successfully imported using the `import.php` script.

Figure 5.4: Simple Mailing List's back-office panel

Next, Reactable Systems' CRON scheduler was configured to work with Simple Mailing List and some minor bugs of Simple Mailing List's code were fixed, also thanks to its excellent support forum.

On March 23rd, 2010, the first Reactable Live! newsletter was successfully sent to more than ten thousands people.

Later, a **custom filter** on mailing list subscribers was added to select groups of users, like press members, museums or musicians.

On September 29th, 2010, a `.pdf` document and a Microsoft Excel 2010 spreadsheet were heavily processed and sorted to retrieve email addresses, names and other informations of more than seven hundreds users; TextWrangler was once again a vital tool.

Chapter 6

Webshop

6.1 Requirements

Reactable Systems needed an e-commerce platform to sell promotional stuff and, in the future, maybe even its main products.

It was immediately obvious that such a project **could not start from scratches**, because it would have required more than an year of work and more than just one developer. It was instead decided to perform a **software selection** and to develop on an existing platform.

To have a better knowledge of what Reactable Systems was looking for in an e-commerce platform, some preliminary **interviews** were made.

The webshop software had to:

- be open source;

- be extendible;

- be very customizable;

- fit the needs of a Small and Medium Enterprise (SME), i.e., not to be over-sized;

- be easy to set up.

- keep its users database separate from WordPress'.

6.2 Software selection

Immediately after the interviews, a web-wide **software selection** was performed, discarding *a priori* paid software, though consciously knowing these usually come with better documentation and customer support.

6.2.1   FreeWebShop.org

FreeWebShop.org (see Figure 6.1) is a free, full featured software that allows to set up an online e-commerce platform within minutes. FreeWebShop.org is written in PHP and uses a MySQL database. It is designed to provide all the features usually needed for a webshop while having an absolute minimal impact on the resources of the server.

FreeWebShop.org 2.2 has been installed and tested.

Most of FreeWebShop.org's queries were displayed in clear in the browser address bar, resulting in an unacceptable security level.

Figure 6.1: FreeWebShop.org

6.2.2   Magento

Magento (see Figure 6.2) is an open source e-commerce web application built on components of the Zend Framework. It was among the first webshops in the U.S. market and is available under three different licenses, one of which is free of charge (named "Community Edition").

Figure 6.2: Magento

Among Magento's several features are:

- Analytics and reporting gears;

- Search Engines Optimization (SEO);

- Mobile devices support;

- Multi-lingual interfaces;

- Import/export of catalogs;

- Multiple payment methods.

Magento 1.4 was not successfully installed due to software issues.

Online reviews on Magento said it required a lot of additional plugins to get satisfactory results and that it also had many performance issues, its benchmark being fifty seconds to add an item.

6.2.3 osCommerce

osCommerce (see Figure 6.3) is a free, GPL-licensed webshop solution that offers a wide range of out-of-the-box features that allows online stores to be set up fairly quickly with ease.

osCommerce 3.0-alpha was not tested due of a specific note on osCommerce website, stating "hosting and service providers [had] to wait until the final release of v3.0 before making it available to their customers".

Though osCommerce has been the market leader for a long time, it resulted too complex for Reactable Systems' needs. Moreover, its source code looked bad designed, which would have resulted in a hard customization process.
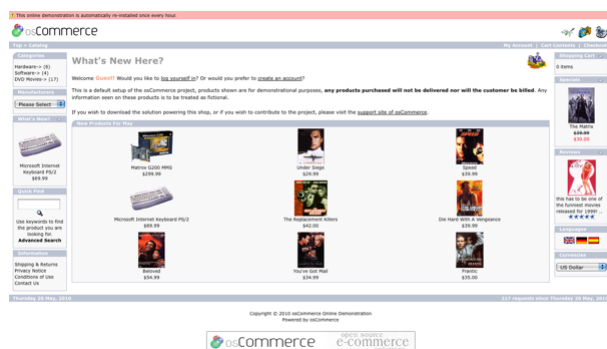

Figure 6.3: osCommerce

6.2.4   PrestaShop

PrestaShop (see Figure 6.4) is an e-commerce solution built in PHP and MySQL, for which many paid or even free modules and themes are available. It also has a multi-lingual support forum and an online wiki.

PrestaShop 1.2 and 1.3 have been installed and tested.

PrestaShop features many payment methods yet-implemented, a well-designed administrator interface and an online shop for modules and themes, named PrestaStore.
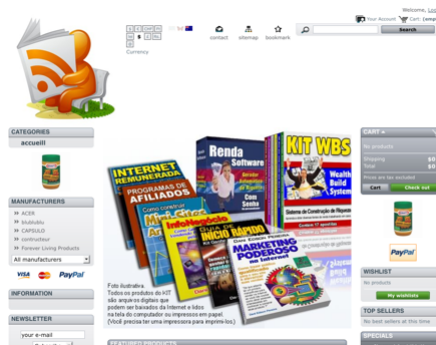

Figure 6.4: PrestaShop

6.2.5   Ubercart

Ubercart (see Figure 6.5) is an open source e-commerce solution fully integrated with the leading open source Content Management System (CMS), Drupal. Ubercart is designed to be enhanced and modified: its core functions can be easily customized and many of the contributed Drupal modules and themes can be ported.
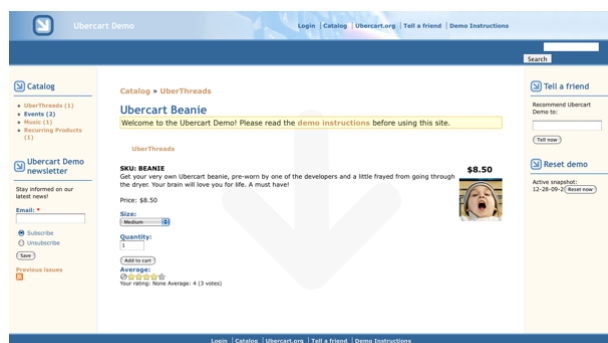

Figure 6.5: Ubercart

No version of Ubercart was tested, because it required Drupal.

Ubercart online demo showed a lot of errors after manual reset. Moreover, its administrator interface looked too simple and poorly designed.

6.2.6   VirtueMart

VirtueMart (see Figure 6.6) is the leading e-commerce solution for the Content Management System (CMS) Joomla!. It is open source, GPL-licensed, written in PHP and extensible via Joomla!'s plugins, components, templates and modules.

No version of VirtueMart was tested, because it required Joomla!.

VirtueMart online demo showed many errors and had an odd pop-up script for full size image displaying.

Figure 6.6: VirtueMart

6.2.7   SmileHouse Workspace

SmileHouse Workspace (see Figure 6.7) is a packaged e-commerce software for integrated e-business solutions. It is based on Java 2 Platform, Standard Edition (J2SE) architecture and offers reliable and easy-to-use solution with low total cost. Workspace is based on reliable technology and open standards. It is widely used and continuously developed and supported by Smilehouse. Workspace requires only small project expenditure and enables a short time-to-market thanks to pre-built components and easy-to-use management interface.

SmileHouse Workspace 1.12 was not successfully installed due to software issues.

Workspace complicated environment settings and badly designed support website made it result too complex for the enterprise needs.
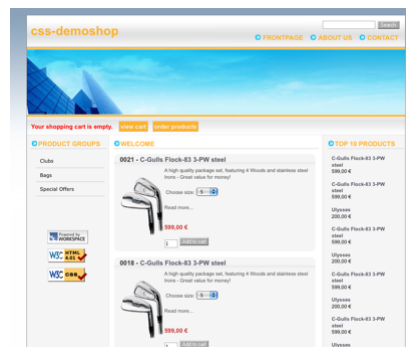
Figure 6.7: SmileHouse WorkSpace

6.2.8   WP e-Commerce

WP e-Commerce (see Figure 6.8) is a shopping cart plugin for WordPress. It
is a web 2.0 application suitable for selling products, services or fees online;
it centers on usability and aesthetics.

WP e-Commerce 3.7 has been installed and tested.

Being a WordPress plugin, WP e-Commerce shares users database with
it, breaking Reactable Systems requirements and neither its source could be
modified to make it use a separate database.
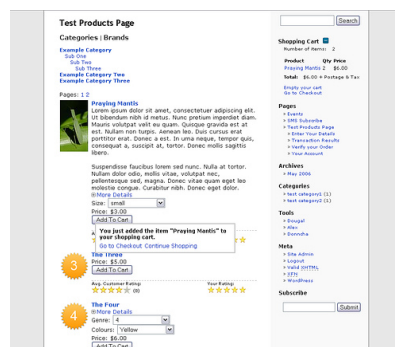
Figure 6.8: WP e-Commerce

6.2.9   Zen Cart

Zen Cart (see Figure 6.9) is a free, user-friendly, open source shopping cart
software. Some of its features are multiple payment options, newsletter man-
agement, free add-ons and an incredibly options-full administrator interface.

Zen Cart 1.3 has been installed and tested.

Figure 6.9: Zen Cart

Zen Cart featured a rough and bad styled administration interface and also looked a little too complex for Reactable Systems needs.

6.3 Development

After a period of closed beta testing, all software but **PrestaShop** turned out not to match Reactable Systems' needs or minimum security and usability standards. It was decided to continue to test PrestaShop in a sandbox, to analyze its features and limits more deeply.

PrestaShop themes and modules make use of both `.php` and `.tpl` (*template*) files for their layout (see Listing 6.1). Template files contain variables that get replaced by user defined output content when PHP web applications (usually the Smarty template engine) parse them.

```
{foreach from=$cmslinks item=cmslink}
  <li class="item"><a href="{$cmslink.link|addslashes}"
    title="{$cmslink.meta_title|escape:'htmlall':'UTF-8'}">
  {$cmslink.meta_title|escape:'htmlall':'UTF-8'}</a></li>
{/foreach}
```

Listing 6.1: blockvariouslinks.tpl (fragment)

6.4 Layout

Plenty of modules and templates for PrestaShop, both charged and free, can be found at its online shop, named **PrestaStore**: some of them were downloaded and tested locally, to have a better view of the possible layouts made available by the community of developers, before starting to manually edit the templates.
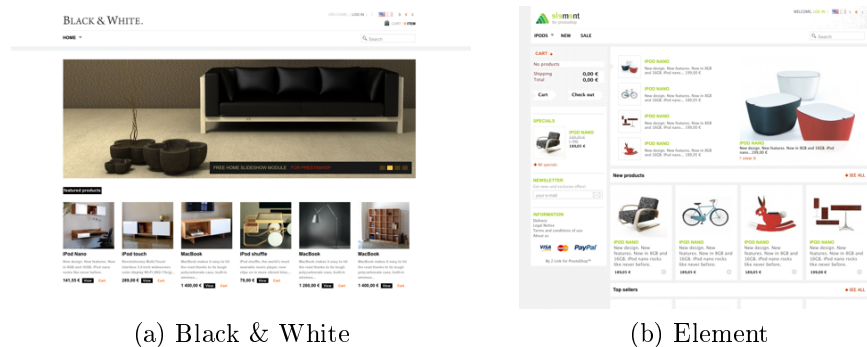
(a) Black & White

(b) Element

Figure 6.10: Prestashop templates

Nevertheless, in the end **no decision** about webshop final template **was taken** by the enterprise.

6.4.1   Black & White

Black & White (see Figure 6.10a) was the first template to be tested: it features a black and white, smooth and very elegant layout, an images sideshow built in JavaScript on the front page and a couple of predefined pages linked in the footer.

Though it was Reactable Systems' favorite, it had quite a lot of hard-coded data in French, very hard to locate and remove, and many custom modules replacing PrestaShop's that were full of bugs.

6.4.2   Element

Element (see Figure 6.10b) was the second template which was downloaded and tested for Reactable Systems webshop. Developed by the same team whom created the Black & White template, it featured a similar top horizontal menu, some nice jQuery animations and, unfortunately, still a bunch of French hard-coded data in its files.

Chapter 7

The Reactable tutorial

7.1 Genesis and requirements

Near the end of the Reactable mobile (see Section 2.4) development process, the enterprise expressed the wish for a **web-based version** of its application; unfortunately, the unpredictable behavior of the web connection, the limited bandwidth and the unknown type of software and hardware the application would run on, make most of the Reactable features **almost impossible to replicate online**. It was thus decided to develop a **tutorial for the Reactable**, to help promoting the upcoming application and to introduce new people to the Reactable world.

The tutorial had to:

- look as close as possible to the original application;

- feature all the Reactable mobile objects;

- include documentation and demo videos;

- be cross browser compatible;

- feature click-driven events;

- be extendible;

7.2 Layout and structure

The tutorial **mirrors the Reactable mobile look and characteristics** (see Figure 7.1): it features the animated table surface and all the original objects, which can be dragged around the screen and dropped on the table

surface, making their semi-transparent controls and documentation appear. A couple of click-driven events are also featured, but objects cannot be rotated like in the Reactable mobile.

7.2.1   Interaction

jQuery 1.4 and **jQuery UI** 1.8 were used to realize the draggable objects and the interactive table: jQuery is a JavaScript library that simplifies HTML document traversing, event handling and Ajax interactions. jQuery UI provides abstraction for low-level interactions and animations, advanced effects and high-level, themeable widgets, built on top of jQuery.

jQuery UI makes defining and using **draggable and droppable objects** an immediate task (see Listing 7.1). Interactions with other dynamic objects are also very easy to manage, thanks to the excellent online demos and documentation.

In the final version of the Reactable tutorial only one object can be dropped onto the table surface at a time, though having more around the screen does not compromise overall performances. When a second object starts dragging, all the ones on the table are returned to their original position.

```
$( 'canvas' ).parent( 'div' ).droppable({
  drop: function() {
    // A draggable object is dropped onto the droppable object
  },
  activate: function() {
    // A draggable object starts dragging
  },
  deactivate: function() {
    // A draggable object stops dragging
  },
  out: function() {
    // A draggable object is dragged out of the droppable object
  },
```

Listing 7.1: tutorial-jquery.js (fragment)

7.2.2   Look and animation

Reactable Systems wanted the tutorial to feature the typical tempo-providing waves and semi-transparent object controls, to make it **look the closest possible to the original Reactable**.

HTML5 features a new tag element named `canvas`, whom provides web developers the ability to *paint* **the screen and to create animations** in a very easy way, also offering backwards compatibility for old browsers.

Using `canvas` it is possible to draw lines, curves or to temporarily stop drawing, move to another point, and start again. Among the plethora of available options are colors, gradients, shadows, paths, images handling, transformation (rotations, translations, scaling) and compositions [24, 25].

Just like real painting, **various drawing layers are created in sequence, one above the other**; to simulate the cyclic behavior of the animation, drawings are combined in one function, looped using JavaScript's `setTimeout()` function, whom calls its argument after a set delay. Last to be drawn are the waves (see Listing 7.2): their speed, opacity and thickness are based on their radius, which changes dynamically and is automatically reset when reaching table half size.

```
function drawWave( canvas, wave_id ) {
  var ctx = canvas.getContext( '2d' );

  if ( canvas.height / 2 < radius[wave_id] )
    radius[wave_id] = default_radius;

  var x =               canvas.width / 2;
  var y =               canvas.height / 2;
  var r =               radius[wave_id];
  var startAngle =      0;
  var endAngle =        2 * Math.PI;
  var anticlockwise =   false;
  var alpha =           1 - r / 180;
  ctx.lineWidth =       r / 8;

  if ( 0 == wave_id )
    alpha = alpha * Math.PI;

  ctx.strokeStyle = 'rgba( 255, 255, 255, ' + alpha + ' )';

  ctx.beginPath();
  ctx.arc( x, y, r, startAngle, endAngle, anticlockwise );
  ctx.closePath();
  ctx.stroke();

  radius[wave_id] = r + canvas.height / ( r + 10 );
}
```

Listing 7.2: tutorial-jquery.js (fragment)

When a draggable object is dropped onto the table surface, its active version, featuring semi-transparent controls, replaces the black and white
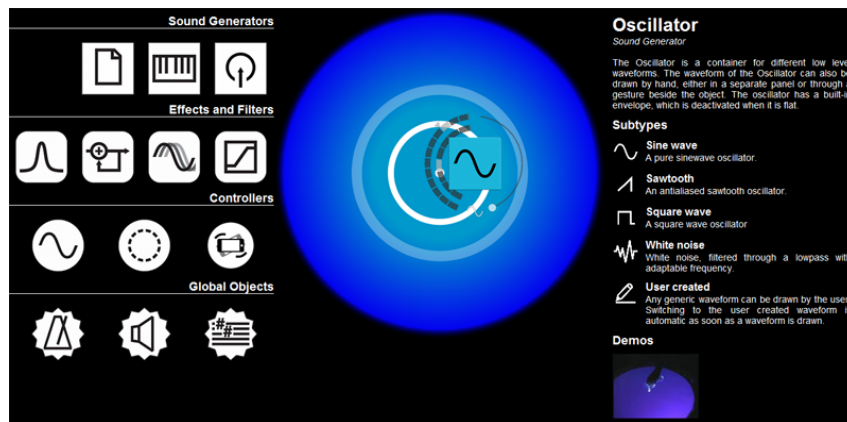
Figure 7.1: The Reactable tutorial

one as long as the object remains on the table. This gets automatically restored when the object leaves.

Thanks to CSS, old browsers users were granted the **same experience** they would have using an HTML5 compatible browser; this because HTML5 is still under development[1] and not all web browsers support it completely nor the same way yet.

For more details on the Reactable tutorial, see Appendix A.

7.3   Documentation

The objects documentation shown in the tutorial is a summarized version of the official online manual for the Reactable mobile: it includes a short object description, a list of its subtypes and, in some cases, a short demo video displayed in a floating box thanks to a jQuery plugin named Fancybox, the same currently in use at Reactable Systems website.

---

[1] http://www.w3.org/TR/html5/

Chapter 8

Conclusions

8.1   Future topics

Talking about a future scenario, an accurate analysis on the **future of bb-Press** is strongly suggested: as announced by John James Jacoby on July 9th, 2010[1], bbPress next official release will be in the form of a WordPress plugin, which does not fit the constraints of the enterprise. Possible alternatives are the search for a replacement software and the start of an independent development branch (either based on the 0.9 or 1.0 branch). On September 11th, 2010[2], Kevinjohn Gallagher, one of bbPress most active contributors, has expressed his intention to start a bbPress fork.

On July 31st, 2010[3], NotOneBit.com project leader has stated **version 2.0 of Simple Mailing List** is "about 70% complete" and has also listed its upcoming features: the most importants are sending mail attachments, creating more mailing lists and sending mails to more lists. An upgrade from version 1.5 should be considered.

Furthermore, the enterprise has expressed the wish to redesign the **back-office architecture** and take some time to consider an improvement of the overall **security** level of its server and the customers data stored in its database. Maintaining a **local copy of the online server** where the website and all of its data is currently located has also been considered, to reduce the downtime in case of updates, crashes or external attacks.

An **improvement of the webshop design**, via custom templates, and its further extension via plugins are also important decisions that should be considered; an alternative option would be the **outsourcing** of the whole e-

---

[1]http://bbpdevel.wordpress.com/2010/07/09/bbpress-as-a-plugin/
[2]http://www.kevinjohngallagher.com/2010/09/i-want-to-fork-bbpress/
[3]http://www.notonebit.com/forum/viewtopic.php?f=32&t=1458

commerce platform project to a third party enterprise, thus resetting the progresses achieved till today and starting its development process from scratches once again.

Finally, the **Reactable tutorial** should be made compatible and accessible via mobile devices, its characteristics should be kept updated with Reactable Systems' new releases and its performance and features should be further improved.

8.2   Final thoughts

During this six months internship several coding styles, development methods and ways of approach changes have been faced. It was an intense working and learning period, in which I had the chance to face the real labor market, filled with problems and hazards: it has surely been **one of the most interesting and instructive times of my life**.

Nevertheless, the young age of the enterprise and its small business experience have played a **central role** in all the decisions about my internship. Several times I had to foresee assignments or to take steps back in my development process because ideas were not clearly defined or minds changed at the very last moment.

Despite this, I would like to thank everyone at Reactable Systems for being so helpful and kind with me. I wish all of them the best of luck.

Appendix A

Reactable tutorial source code

The Reactable tutorial jQuery script is here listed and commented.

A.1 Calling jQuery

jQuery is called using a special handler **to avoid namespaces difficulties**, as explained on its official website[1].

```
jQuery( document ).ready( function($) {
```

A.2 Droppable objects and related events

The `canvas`'s parent `div` element is defined as a **droppable object**.

```
$( 'canvas' ).parent( 'div' ).droppable({
```

Droppable-specific events can now be bound.

Whenever **a draggable object is dropped onto the droppable object**, the image element it contains is hidden, if it was visible.

If there is no dropped object or if there is any and the dragged one is not among them, it is added to the dropped objects list.

To **prevent multiple events binding**, click and double-click events get first unbound from the dragged object and then bound again: to distinguish between the two events, the former is delayed via JavaScript's `setTimeout()` function, which return value is stored along with dragged object id in two local variables. On the other side, double-click event resets this variables before calling the associated function.

---

[1]http://api.jquery.com/ready/

```
drop:            function( event , ui ) {
  var draggable_id =  ui.draggable.attr( 'id' );
  var draggable_img = ui.draggable.find( 'img' );

  if ( draggable_img.is( ':visible' ) )
    draggable_img.hide();

  if (
    0 == dropped_objects.length
    || ( 0 < dropped_objects.length
      && −1 == dropped_objects.indexOf( draggable_id ) )
  )
    dropped_objects.push( draggable_id );

  ui.draggable.unbind( 'click' ).unbind( 'dblclick' );

  ui.draggable.click( function() {
    clicked_obj_id = draggable_id;
    clicktimer =      setTimeout( checkSingleClick , 200 );
  }).dblclick( function() {
    clicked_obj_id = null;
    clearTimeout( clicktimer );

    doubleClickAction( draggable_id );
  });
},
```

Whenever **a draggable object starts dragging**, if there is any dropped object and if the dragged one is not among them, standard information message and all dropped objects original position, but the dragged one, are restored.

Then click and double-click events get unbound from non-dropped objects.

```
activate:     function( event , ui ) {
  var draggable_id = ui.draggable.attr( 'id' );

  if ( 0 < dropped_objects.length
    && −1 == dropped_objects.indexOf( draggable_id ) ) {
    resetInfo();
    resetDroppedObjects( draggable_id );
  }

  unbindNotDropped();
},
```

Whenever **a draggable object stops dragging**, click and double-click events get unbound from non-dropped objects.

```
deactivate: function() {
  unbindNotDropped();
},
```

Whenever **a draggable object is dragged out of the droppable object**, its index in the dropped objects list is retrieved and the visibility of the image element it contains is restored, if it was hidden.

If dropped objects list contains the dragged object, the array gets spliced to remove it from the list.

Last, standard information message is restored.

```
out:            function( event, ui ) {
  var draggable_id =    ui.draggable.attr( 'id' );
  var draggable_img =   ui.draggable.find( 'img' );
  var draggable_index = dropped_objects.indexOf(
    draggable_id );

  if ( draggable_img.is( ':hidden' ) )
    draggable_img.show();

  if ( -1 != draggable_index )
    dropped_objects.splice( draggable_index, 1 );

  resetInfo();
},
```

Last, the *tolerance* parameter is set to `fit`, meaning draggable objects have to overlap the droppable one **entirely** to be considered as dropped.

```
  tolerance:    'fit'
});
```

A.3 Draggable objects

Draggable objects container size is based on browser window size. The `-4` pixels correction is mandatory tweak for a few web browsers.

```
$( '#objects-container' ).css({
  'height': $( window ).height() - 4,
  'width':  $( window ).width()
});
```

The DOM tree is browsed to find specific `li` elements: these are defined as **draggable objects** and are bound inside a container (previously defined), do not duplicate themselves when dragged and stack on each other.

```
$( '#objects > li > ul > li ' ).draggable({
  containment: '#objects−container ',
  helper:       'original ',
  stack:        '.ui−draggable '
});
```

A.4   Image pre-loading

Since `canvas` elements create images objects on the fly, **image files must be pre-loaded** in cache to avoid errors during real-time execution.

The `preloadControls()` function instantiates the *imageArray* array and selects all draggable objects: for each of them, the function adds its draggable controls image to the images list.

```
preloadControls();

function preloadControls() {
  document.imageArray = [];

  $( '.ui−draggable ' ).each( function( index ) {
    document.imageArray[index] =      new Image;
    document.imageArray[index].src = 'images/controls/'
      + $( this ).attr( 'id ' ) + '.png';
  });
}
```

A.5   Click-driven events

Before executing the single click function, it is important to **check if user single- or double-clicked** on the object: if the *clicked_obj_id* variable is set, single-click timer is reset and associated function is executed.

```
var clicktimer;
var clicked_obj_id;

function checkSingleClick() {
  if ( clicked_obj_id ) {
    clearTimeout( clicktimer );

    singleClickAction( clicked_obj_id );
  }
}
```

In case of a double-click event, standard information message and all dropped objects position are restored.

```
function singleClickAction( obj_id ) {
  showInfo( obj_id );
}

function doubleClickAction( obj_id ) {
  resetInfo();
  resetDroppedObjects();
}
```

The `showInfo()` function receives an object id as parameter, loads the associated file and injects its code into the DOM element with class *info*. When it is done, a **callback function** is executed: it binds a click event to every anchor inside the just-inserted code, instantiating a Fancybox element for YouTube videos (as described on its official website[2]), and then it prevents event default behavior.

```
function showInfo( obj_id ) {
  $( '.info' ).load( 'docs/' + obj_id + '.html', function() {
    $( '.info > a' ).click( function( event ) {
      $.fancybox({
        'autoScale':      false ,
        'href':           this.href.replace( new RegExp(
          "watch\\?v=", 'i' ), 'v/' ) ,
        'orig':           $( this ) ,
        'padding':        0 ,
        'transitionIn':   'elastic' ,
        'type':           'swf'
      });

      event.preventDefault();
    });
  });
}
```

The `resetInfo()` function resets the standard message of the DOM element with class *info*.

```
function resetInfo() {
  $( '.info' ).html(
    '<h2>Drag and drop an object onto the Reactable.</h2>' );
}
```

The `resetDroppedObjects()` receives an object id as parameter, checks if there is any dropped object and, if the passed one is not among them, it

---

[2]http://fancybox.net/blog

cycles through them: it pops out the dropped objects array an id, retrieves the associated object and finds the image element inside it.

Then image visibility is restored, if it was hidden, as is its original position, thanks to jQuery's `animate()` function.

```
function resetDroppedObjects( obj_id ) {
  if ( 0 < dropped_objects.length
    && -1 == dropped_objects.indexOf( obj_id ) )
    while ( 0 < dropped_objects.length ) {
      var dropped_obj_id = dropped_objects.pop();
      var dropped_obj =    $( '#' + dropped_obj_id );
      var bw_img =          dropped_obj.find( 'img' );

      if ( bw_img.is( ':hidden' ) )
        bw_img.show();

      dropped_obj.animate(
        { left: 0, top: 0 },
        600
      );
    }
}
```

The `unbindNotDropped()` function selects all draggable objects and, for each of them, checks if there is no dropped object or if there is any and the current one is not among them; if so, click and double-click events get unbound from it.

```
function unbindNotDropped() {
  $( '.ui-draggable' ).each( function() {
    if (
      0 == dropped_objects.length ||
      ( 0 < dropped_objects.length
        && -1 == dropped_objects.indexOf(
          $( this ).attr( 'id' ) ) )
    )
      $( this ).unbind( 'click' ).unbind( 'dblclick' );
  });
}
```

A.6   Drawing functions

Drawings are combined in a single function, named `startReactable()`.

The `startReactable()` function gets the `canvas` element specifying the array index because this is going to be used by standard JavaScript functions; then it **checks if the browser is HTML5 compatible** before proceeding

with the execution. The function gets `canvas` context, resets it and calls the drawing functions: the waves functions beyond the first one are called only if the iterations counter has got to a specific value.

The function then checks if there is any dropped object and it cycles through them drawing their controls.

The iterations counter is incremented (only if necessary) and then the function is called again after a delay, simulating a `setInterval()` function call.

```
startReactable();

function startReactable() {
  var canvas = $( 'canvas' )[0];

  if ( canvas.getContext ) {
    var ctx = canvas.getContext( '2d' );

    ctx.clearRect( 0, 0, canvas.width, canvas.height );
    drawBg( canvas );
    drawWave( canvas, 0 );

    if ( 18 < iterations ) {
      drawWave( canvas, 1 );

      if ( 36 < iterations ) {
        drawWave( canvas, 2 );

        if ( 54 < iterations )
          drawWave( canvas, 3 );
      }
    }

    if ( 0 < dropped_objects.length )
      for ( var i in dropped_objects )
        drawControls( canvas, dropped_objects[i] );

    if ( 54 >= iterations )
      iterations++;

    setTimeout( startReactable, interval );
  }
}
```

The `drawBg()` function receives a `canvas` element as parameter and retrieves its context. Note that there is no check on browser HTML5 compatibility here: this because main function **already performs it** before calling this function.

The function **paints canvas background** using a radial gradient: its size is based on `canvas` size and the colors blend are the same of the original Reactable.

```
function drawBg( canvas ) {
  var ctx = canvas.getContext( '2d' );

  var x1 = x2 = r2 = canvas.width / 2;
  var y1 = y2 =       canvas.height / 2;
  var r1 =            100;
  var radgrad =       ctx.createRadialGradient(
    x1, y1, r1, x2, y2, r2 );

  radgrad.addColorStop( 0,   '#09c' );
  radgrad.addColorStop( 0.9, '#0404f0' );
  radgrad.addColorStop( 1,   'rgba( 0, 0, 0, 0 )' );
```

The context size is based on `canvas` size and its color is the just-created radial gradient. The `fillRect()` function does the painting.

```
  var left = top = 0;
  var width =       canvas.width;
  var height =      canvas.height;
  ctx.fillStyle =   radgrad;

  ctx.fillRect( left, top, width, height );
```

Last is the **central point drawing**: it is painted as a tiny white circle, right at the center of the `canvas`.

It is **mandatory** to wrap the `arc()` function inside a `canvas` path, since main function executes several drawings which must remain separate.

```
  var x_p =              x1;
  var y_p =              y1;
  var r_p =              7;
  var startAngle =       0;
  var endAngle =         2 * Math.PI;
  var anticlockwise = false;
  ctx.fillStyle =        '#fff';

  ctx.beginPath();
  ctx.arc( x_p, y_p, r_p, startAngle, endAngle,
    anticlockwise );
  ctx.closePath();
  ctx.fill();
}
```

The `drawWave()` function receives a `canvas` object and an id as parameters. It **paints the waves** on the `canvas`.

The function retrieves the `canvas` context and, if wave radius grew over half `canvas` size, resets it to its default value. Note that there is no check on browser HTML5 compatibility here: this because main function **already performs it** before calling this function.

Wave is positioned at `canvas` center and its radius is retrieved from a local array variable using the id passed to the function as index; wave radius also determines its opacity and thickness. Moreover, leading wave opacity is decreased slower than the others.

Once drawn the wave, its radius is incremented and saved on the local array variable.

```
function drawWave( canvas, wave_id ) {
  var ctx = canvas.getContext( '2d' );

  if ( canvas.height / 2 < radius[wave_id] )
    radius[wave_id] = default_radius;

  var x =               canvas.width / 2;
  var y =               canvas.height / 2;
  var r =               radius[wave_id];
  var startAngle =      0;
  var endAngle =        2 * Math.PI;
  var anticlockwise = false;
  var alpha =           1 - r / 180;
  ctx.lineWidth =       r / 8;

  if ( 0 == wave_id )
    alpha = alpha * Math.PI;

  ctx.strokeStyle = 'rgba( 255, 255, 255, ' + alpha + ' )';

  ctx.beginPath();
  ctx.arc( x, y, r, startAngle, endAngle, anticlockwise );
  ctx.closePath();
  ctx.stroke();

  radius[wave_id] = r + canvas.height / ( r + 10 );
}
```

The `drawControls()` function receives a `canvas` object and an object id as parameters. It **draws controls image** for the dropped object with id as the passed one.

The function retrieves `canvas` context and the object with id as the passed one. Note that there is no check on browser HTML5 compatibility here: this because main function **already performs it** before calling this function.

The controls image position is based on dropped and droppable object

positions.

Once instantiated the *Image* object, the function sets its source and draws it onto the `canvas`.

```
function drawControls( canvas, obj_id ) {
  var ctx = canvas.getContext( '2d' );

  var object =              $( '#' + obj_id );
  var x =                   object.position().left
    − $( '.ui−droppable' ).position().left − 31;
  var y =                   object.position().top
    − $( '.ui−droppable' ).position().top − 18;
  var height = width = 160;
  var img =                 new Image();
  img.src =                 'images/controls/' + obj_id + '.png';

  ctx.drawImage( img, x, y, width, height );
}
```

A.7   Local variables

Outside the jQuery `ready()` function, a few local variables are defined: waves default radius, dropped objects list, `canvas` refresh rate, iteration counter and waves radius array.

```
});

var default_radius =   30;
var dropped_objects = [];
var interval =         1000/60;
var iterations =       0;
var radius =           [default_radius, default_radius,
  default_radius, default_radius];
```

Bibliography

[1] R. S. Pressman, *Principi di Ingegneria del Software*. McGraw-Hill, 2004.

[2] "WordPress coding standards."
http://codex.wordpress.org/WordPress_Coding_Standards.

[3] M. Kaltenbrunner, S. Jordà, G. Geiger, and M. Alonso, "The reacTable*
A Collaborative Musical Instrument," 2006.

[4] M. Puckette, "Max at seventeen," *Computer Music J.*, vol. 26, no. 4,
pp. 31–43, 2002.

[5] S. Jordà, M. Kaltenbrunner, G. Geiger, and R. Bencina, "The re-
acTable*," 2005.

[6] S. Jordà, G. Geiger, M. Alonso, and M. Kaltenbrunner, "The reacTable
Exploring the Synergy between Live Music Performance and Tabletop
Tangible Interfaces," in *Conference on tangible and embeded interaction*,
2007.

[7] S. Jordà and M. Alonso, "Mary had a little scoreTable* or the reacTable*
goes melodic," in *New Interfaces for Musical Expression (NIME)*, 2006.

[8] S. Jordà, *Digital Lutherie Crafting musical computers for new musics'
performance and improvisation*. PhD thesis, Ph.D. Dissertation. UPF,
2005.

[9] S. Jordà, "Instruments and Players Some thoughts on digital lutherie,"
Journal of New Music Research, vol. 33, 2005.

[10] R. Bencina, M. Kaltenbrunner, and S. Jordà, "Improved Topological
Fiducial Tracking in the reacTIVision System," 2005.

[11] S. Jordà, "Multi-user Instruments Models, Examples and Promises,"
2005.

[12] S. Jordà, "Digital Instruments and Players Part II: Diversity, Freedom and Control," in *International Computer Music Conference*, 2004.

[13] M. Kaltenbrunner, G. Geiger, and S. Jordà, "Dynamic Patches for Live Musical Performance," 2004.

[14] S. Jordà, "Digital Instruments and Players Part I: Efficiency and Apprenticeship," 2004.

[15] S. Jordà, "Interactive music systems for everyone exploring visual feedback as a way for creating more intuitive, efficient and learnable instruments," 2003.

[16] S. Jordà, "Sonigraphical Instruments From FMOL to the reacTable*," 2003.

[17] S. Jordà, "FMOL Toward User-Friendly, Sophisticated New Musical Instruments," *Computer Music Journal*, vol. 26, pp. 23–39, 2002. Software implementations.

[18] S. Jordà and O. Wüst, "FMOL A System for Collaborative Music Composition over the Web," 2001.

[19] S. Jordà, "New Musical Interfaces and New Music-making Paradigms," 2001.

[20] Organisation for Economic Co-operation and Development, *The Future Digital Economy: Digital Content Creation, Distribution and Access.* 2006.

[21] Organisation for Economic Co-operation and Development, *Towards a knowledge-based economy.* 2001.

[22] D. Kusek and G. Leonhard, *The Future of Music: Manifesto for the Digital Music Revolution.* 2005.

[23] M. Leman, F. Avanzini, A. de Cheveigné, and E. Bigand, "The societal contexts for sound and music computing: research, education, industry, and socio-culture," *J. New Music Res.*, vol. 36, pp. 149–167, Sept. 2007.

[24] "Mozilla.org's canvas tutorial." https://developer.mozilla.org/en/Canvas_tutorial.

[25] "Dive into HTML5's canvas tutorial." http://diveintohtml5.org/canvas.html.

[26] "Apache HTTP server documentation."
http://httpd.apache.org/docs/2.2/.

[27] "MySQL reference manual."
http://dev.mysql.com/doc/refman/5.0/en/.

[28] "PHP.net manual."
http://www.php.net/manual/.

[29] "HTML.it's PHP guides."
http://php.html.it/guide/.

[30] "WordPress codex."
http://codex.wordpress.org/.

[31] "Unofficial PHP cross reference of bbPress."
http://phpxref.ftwr.co.uk/bbpress/.

[32] "Wikipedia, the free encyclopedia."
http://wikipedia.org/.

[33] "WikiBooks' LaTeX guide."
http://en.wikibooks.org/wiki/LaTeX.

[34] "GUIT (Gruppo Utilizzatori Italiani di TeX) forum."
http://guit.sssup.it/phpbb/.