Master Thesis
by Dariusz Korsak
Supervised
by dr Ryszard JAKUSZEWSKI

SCADA SYSTEMS implementation in www applications
Presentation Overview

- Thesis Premises
- Internet programming
- iFix SCADA application
- iFix communication with external applications
- Implementation
- Conclusions
The goal of this Thesis is to develop web application integrated with SCADA system for automated production. It consists of fully expanded e-commerce application with CMS SYSTEM and SCADA synoptic panel (Human-Machine Interface) which allows to control production process with a web browser.
**PHP** (recursive acronym: PHP: Hypertext Preprocessor) is an HTML (XHTML)-embedded open source high level scripting language which allows web developers to write dynamically generated pages.

PHP runs on a web server, taking PHP code as its input and creating Web pages as output. It can be considered as an alternative to: Microsoft's ASP.NET system, Sun Microsystems' Javascript, Cold Fusion or Perl.

PHP can be used with a large number of relational database management systems (MySQL, Oracle, PostgreSQL, SQLite) or use alternate ways of storing data like text or XML files. It runs on majority of operating systems and on most popular web servers and it’s widely used across the internet (over 18 million Internet domains are currently hosted on servers with PHP installed).
MySQL - MySQL is most popular engine of relational database management system (RDMS). It finds great application in internet projects thanks to its speed and efficiency. It is available for almost all systems and architectures.

Features:
- It’s written in C and C++ (very good efficiency)
- Transactions support
- Large number of supported data types
- Each column can be suited to the given size of data.
- Support for grouping and aggregating clauses
- Easy from administrator point of view security system, passwords encoding
- Connection to server by: TCP/IP, ODBC
- Many different table (relation) types suited for different applications:
  - MyISAM – default, does not support transactions
  - MEMORY – the fastest, everything is stored in RAM memory,
  - InnoDB – supports transactions, referential integrity constraints, and higher concurrency.
  - CSV – stores the data in standard CSV files
XHTML (Extensible HyperText Markup Language) is a successor of no longer developed HTML language that has the same expressive possibilities as HTML, but a stricter syntax. XHTML code is not anyhow parsed or compiled, it is send as is to the browser where it is interpreted.

Main difference in respect to HTML is that XHTML can be only a carrier of text, structure like headers, paragraphs, lists, links, etc., and information about include files like java scripts, graphics, style sheets. It does not carry the information about the outlook, presentation of the document, that information is moved to the external CSS (Cascading Style Sheets) file.

CSS is the list of directives (rules) determining how the document should be rendered by the browser. In CSS file one can describe all the rules responsible for presentation of a document like fonts family, text color, margins or even position of given element in respect to other elements or browser’s window. Moving that description into external file enables possibility of easy changing formatting of multiple documents. Since the CSS content is kept in browser’s cache significantly less data is exchanged between server and client (browser) so the web page can be loaded faster.
JS – javascript - script language - a system of programming codes, created by Netscape, that can be embedded into the HTML of a web page to add functionality. JavaScript should not be confused with the Java programming language. In general, script languages such as JavaScript are easier and faster to code than more structured languages such as Java and C++.

AJAX (Asynchronous JavaScript and XML) is not as it is often considered a new technology, language or library but rather a new way of usage of existing technologies like:
- HTML or XHTLM, CSS – for standard layer of presentation
- Document Object Model (DOM) – for dynamic content display
- XMLHttpRequest – for asynchronous data transfer
- JavaScript – binds everything together

AJAX provides sending requests from www application to the server and receiving only necessary data. Data between AJAX and server can be exchanged in XML standard and next read by JavaScript on client’s side. In easy words to introduce some changes on the web site (like entering some text to the form, clicking some links) there is no need for each time reloading it to see the result.

That solution provides better speed and reaction time of web applications and makes them more interactive. Only small portions of data are exchanged between client and server so the time between interaction and display of results is small. This creates opportunity that www applications can become more similar to the desktop ones. Data is sent asynchronously so the application is not hold until it receives response from the server.
Data transfer

Classic web application model

- User interface
- JavaScript call
- HTML/CSS data
- HTTP request
- HTML/CSS data
- Web server
- Datastores, backend processing, legacy systems
- Server-side systems

Ajax web application model

- User interface
- JavaScript call
- HTML/CSS data
- HTTP request
- XML data
- Web and/or XML server
- Datastores, backend processing, legacy systems
- Server-side systems
Ajax example

1. User uses web browser to access index.html.
2. Web browser makes normal HTTP request to web server requesting index.html.
3. Server responds to the call by sending back index.html and the JavaScript file quickstart.js that is referenced in index.html.

4. Client loads index.html and quickstart.js and composes the page on the screen.

5. On the page, the user starts typing his/her name.

6. Quickstart.js makes an asynchronous call to quickstart.php on the server on behalf of the user.

7. Quickstart.php is executed on the server and returns the results in XML format.

8. Quickstart.js receives response from the server and uses the data to update user's display.

9. User's page is updated with new data while user continues working on the page.
iFix SCADA Application

“Komponowanie farb” it is an application that allows to control the industrial process of paint composing. Its functions provide production of specified amount of paint according to the chosen recipe, control over production process and reports generation.
Production process stages:

• Filling — in this stage the tank is filled with specified composition of paint ingredients, the input valves are opened, mixing is turned on, the liquid level in the tank is rising.

• Heating — in this stage the reactor is no longer filled, so the respective valves are closed, steam is delivered to the reactor and the valve is open, red lights on the reactor are indicating that the heating process is in progress.

• Thermal Annealing — valves are open in the way to keep constant temperature of the reactor.

• Cooling — after some time of thermal treatment the cooling stage is started, the temperature of the reactor decreases, input valves of cooling ingredient are opened, blue indicators in the reactor are blinking.

• Emptying — the last stage, the reactor is emptying and the ready product is going to the containers.
ifix communication with external applications

Ifix application requires access to Microsoft Access database in order to collect information about placed orders and to communicate with the internet module. Communication with external database can be realized directly by Visual Basic ADO components or by ODBC server.
Schema of communication in the system
The access to the process variables of the iFix application is realized through the ODBC server and special function blocks SQT and SQD. SQT blocks are triggers that are executed in cycles in some period of time or controlled by specified events.

In order to write data from iFix database to external database (and opposite) another SQD block need to be executed. In SQD block variables to or from which the information will be read or write are defined. iFix application saves process variables every second to the external database, so the current process state can be available to the browser client and the control can be realized from the internet browser as well.
Algorithm for assigning process variables

Start

Value of panel variable from Access database is assigned to the panel variable in iFix internal database.

No

panel -- 1

Yes

Process variables values from iFix database are assigned to variables in iFix database

Process variables values from Access database are assigned to the Access database

Value 0 is assigned to the panel variable in Access database

End
Implementation

Web shop

E-commerce in present times creates so called “New Economy” – economy of global range in which country borders do not exist.

Polish e-commerce market is now one of the most dynamically growing and perspective market in Europe with sale over 3 billion zlotys in year 2005 and reached level of 1% of whole retail sale in Poland[1]. The most common form of e-commerce in internet is the web shop.

The application which is the subject of this thesis tries to fit into this trend and offer all the functionalities that are necessary to provide convenient user service.

# Web shop registration form

<table>
<thead>
<tr>
<th>Personal details</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Name: *</td>
<td></td>
</tr>
<tr>
<td>Surname: *</td>
<td></td>
</tr>
<tr>
<td>Email: *</td>
<td></td>
</tr>
<tr>
<td>Tel number: *</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Company details</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Firm name:</td>
<td></td>
</tr>
<tr>
<td>TIN:</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Address</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Street:</td>
<td></td>
</tr>
<tr>
<td>Street no.:</td>
<td></td>
</tr>
<tr>
<td>Apartment no.:</td>
<td></td>
</tr>
<tr>
<td>ZIP code:</td>
<td></td>
</tr>
<tr>
<td>City:</td>
<td></td>
</tr>
<tr>
<td>Country:</td>
<td></td>
</tr>
</tbody>
</table>

- Above address is also an address for shipping: **[ ]**

<table>
<thead>
<tr>
<th>Login &amp; Password</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Login: *</td>
<td></td>
</tr>
<tr>
<td>Password: *</td>
<td></td>
</tr>
<tr>
<td>Confirm password: *</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Additional information</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>I confirm that I have read and agreed to the Terms and conditions when buying products in this shop: <strong>[ ]</strong></td>
<td></td>
</tr>
</tbody>
</table>

* All fields with a star must be filled.
# Web shop shopping cart

<table>
<thead>
<tr>
<th>Photo</th>
<th>Name</th>
<th>Quantity</th>
<th>Price brutto</th>
<th>Delete</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image.png" alt="Photo" /></td>
<td>Paint 1</td>
<td>15</td>
<td>75.00 $</td>
<td>Delete</td>
</tr>
</tbody>
</table>

**Total order price:**

- 10% discount for shopping above 20 zł
- Special delivery
- Shipping is 10% cheaper

<table>
<thead>
<tr>
<th></th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>75 $</td>
<td></td>
</tr>
<tr>
<td>-7.50 $</td>
<td></td>
</tr>
<tr>
<td>+100.00 $</td>
<td></td>
</tr>
<tr>
<td>-20.00 $</td>
<td></td>
</tr>
</tbody>
</table>

**Total price to pay:** 147.5 $

[Clear Cart] [Checkout]
Web shop order confirmation

**Order number:** #3

Thank you for your shopping. We hope you'll visit us again soon.

<table>
<thead>
<tr>
<th>Photo</th>
<th>Name</th>
<th>Quantity</th>
<th>Price brutto</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Paint 1" /></td>
<td>Paint 1</td>
<td>Quantity: 15</td>
<td>75.00 $</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Order number:</th>
<th>#3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total cost including shipping:</td>
<td>147.50 $</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shipping form</th>
<th>Special delivery 80 $</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order status:</td>
<td>waiting for confirmation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Client:</th>
<th>Ship address</th>
</tr>
</thead>
</table>
| Dariusz Korsak  
Street 3 / 5  
21-542 Gliwice | Dariusz Korsak  
Street 3 / 5  
21-542 Gliwice |
| cordo@1000lecie.pl  
235-235-23 | |
Content Management System is an application that provides easy managing over www site. Almost all aspects of a web site can be controlled by the CMS with use of easy interfaces.
CMS – Orders module

Administrator of the system has possibility to change the order status. When the status is changed to confirmed the ordered products are passed to the SCADA Panel module where their production process can be initiated.
One of the most important goals of this project was to create working and first of all functional synoptic panel that will allow controlling an industrial process from the web page. Up till now it was very difficult to create such an application that would equal the demands of reliability and usability. Mainly because the real-time processes has to be presented and controlled in non real-time environment. But recently with development of new technologies like XMLHttpRequest (asynchronous data transfer) or Document Object Model (dynamic content display) it is not longer impossible.
Rector visualization

Synoptic panel view is composed of the reactor graphical representation and panel menu.

On the reactor representation one can observe the visualization of the production process. In the panel menu the information about paint level in the tank, temperature, and the process stage is presented. Currently running process is marked by a red control, for inactive process the control is grey.
Synoptic panel

- The synoptic panel operator can monitor but also control the production process. Controls in panel menu allow turning on or off respective process stages, valves can be opened or closed so as the mixer.

- All the changes of the process variables and the reactor visualization are visible instantly and without refreshing of the page. The panel operator has the impression of real-time operation of the application. In fact each second with use of AJAX (Asynchronous JavaScript and XML) the HTTP request takes the form of a JavaScript and is sent asynchronously (without pausing user’s interaction with the web application) to the server. Next the script checks if the response is finished and if it is the response is returned. The effect is that the data is gathered from the server database and presented on the screen every second.
Conclusions

- In this Master Thesis the following elements were done:
  - Web Shop application with functionalities like:
    - User accounts creation and edition
    - Individual shopping cart for each user
    - Login panel
    - Newsletter module
    - Management of orders
    - News
    - Promotions
    - Rebates
    - Support for many languages
  - Control Management System with modules:
    - Administrator account management
    - Domain description
    - Language panel
    - Modules responsible for site presentation (categories, navigational menus, products)

- The result fulfills that task. It provides such production mechanism that is strictly connected with current requisition, which enables to ideally fit the quantity and type of production to the demands of the market. In this work the production which was controlled, was the industrial process of paint composing, however the area of potential application is unlimited, especially in case when the cost of production of single product is very high.

- The module for remote production supervision through the web browser provides possibility of almost real time observation of the process, however remote control over internet or intranet in not fully certain. It is because of that type of connection can be treacherous (lost packages, connection interruption and dependence on external factors). That is why that type of solution is not very often used in direct control over devices and machinery.

- On the other hand large flexibility of that solution and also its compatibility with different system platforms, and finally low costs of implementation and easiness of access enables creation of complicated and distributed systems of supervision and reporting.