

CV of Damir Vukres

Keywords relating to my past work:

- ✓ Web developer,
- ✓ web architect,
- ✓ web service architect/developer,
- ✓ C/C++ programmer,
- ✓ database designer,
- ✓ lead developer,
- ✓ product development manager.

Recently I'm working on:

- Data mining
- ETL
- Business intelligence

solutions using PHP, Prolog, Java and C/C++.

I've started with programming at early eighties of last century, almost 30 years ago. Assembler with , at that time very popular , Z80 processor, and basic interpreter as programming language were my first software points of interest.

At late eighties I've started with C, late nineties with C++, STL about 2003.

Example from that time (Visual C++ , MFC):

www.w-app.net/BetMaster.zip

BetMaster is helper application for players that want to know where they put their money on, and may need to know possible changes before really decide what to do. Engine is in DLL.

After 2005. PHP (better to say LAMP with supporting technologies, then JavaScript in first place, and also XML, XHTML, CSS, Ajax, REST, etc...) is my principal occupation.

Applications Examples:

- **OB CMS (One Board)** is one screen (board) CMS with responsive design.

Front end: j.w-app.net/index.php

Admin credentials on demand.

Screenshots: www.w-app.net/CV/OBCMS/



- At **BB (Basic Business)** project for USA client, there are a lot of dhtml examples. **BB** deals with quotes, invoices, inventory, inputs and outputs, products, workers, working hours, partners, etc...

www.kompenzacija.net/Kohler

Feel free to use this demo version and examine all of described examples.

- Some kind of different approach has **Sync** project, web service developed for one Slovenian client. It uses rsync for incrementally synchronisation. **Sync** helps all employers of one project studio to online synchronise their parts of projects from their computers.

www.kompenzacija.net/Sync

At this demo synchronisation itself is not visible because it is not possible (well, it is, but is repulsive complicated) to view all states before and later. It just shows how it handles the task.

- o **iBooking's** frontend:

<http://ibooking.w-app.net/>

has something interesting in that matter: look at calendar. It is possible to click on some day, and it will become green. First day and last day of chosen interval are half-green, others are full-green. Same for occupied accommodation: half-red and full-red. The even action (maximum clicked, or months at any adjust that users

March / 2017							April / 2017						
M	T	W	T	F	S	S	M	T	W	T	F	S	S
		1	2	3	4	5						1	2
6	7	8	9	10	11	12	3	4	5	6	7	8	9
13	14	15	16	17	18	19	10	11	12	13	14	15	16
20	21	22	23	24	25	26	17	18	19	20	21	22	23
27	28	29	30	31			24	25	26	27	28	29	30

may chose (and vice versa). This functionality takes about 18 Kb of JavaScript.
We could proceed with more ajax/dhtml fun: pick up some interval on calendar by clicking on it, or use drop downs, and click submit. If chosen period of time is available for booking, the button for detail page will appear. Ajax will bring prices, discounts and/or up-front amount, but You have possibility to change your mind. You may choose another number of persons, or, by drops, change time period. At any case ajax will react with new prices/amounts situations. Additional services (bunch of dhtml code) are independent against all of that, and could be clicked at any time unrelated to ajax matters, but their amounts will be added to ajax calculated total, and all of that will be something what user can see on next page as well as on his email in WYSWYG manner, which means that email will contain the same graphic and html.

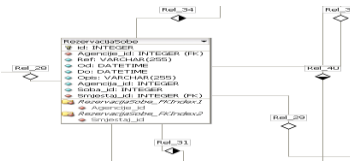
Interesting solution for creating different data inputs frontends from backend.

Admin interface: <http://ibooking.w-app.net/Admin/>

Admin credentials on demand.

This is first draft of database for **iTour** project:

<http://www.w-app.net/CostaDraft1.png>



- o **iBooking, iTour**

Among my own projects, **iBooking** has been improved several times. This is one of its early versions:

<http://www.w-app.net/WebApartmani.pdf>

Note: above link is document in Croatian language. Since I do not have customers for that product out of Croatia, I did not translate this user manual. But, in purpose of getting impression about software itself, this 40 pages could help even You cannot read it.

- o **iShares**

<http://ishares.w-app.net/>

iShares allows shareholders to to gain insight their project.

Two level admin backoffice. Some interesting security issues...

Admin interface: <http://ishares.w-app.net/Client/index.php>

Admin credentials on demand.



- o **Tree**

<http://www.w-app.net/tree/stablo.php>

Decision tree, generating and using

- o **Logger**

This is more then just notifying of interesting moments during program execution. It may replace unit testing, send email or even sms to person in charge, prevent bugs or treat them when they arise, track multiple threads, etc...

This UML diagram:

www.w-app.net/LOGGER.png

shows core logging engine, and just file related observers because of clarity, easy reference and quantity of information. Others (database and email related, as well as actions hunters-responders, or logger supported error recovering) follow the same principles.

Observers:

www.w-app.net/LoggerObservers

I can say to any of them what to do, or not to do, at which file, method, class, line, and at what way to perform what I expect from them:

- **Users logg browser**

Screenshots: www.w-app.net/CV/ULB.pdf

Application for users administration and logs control. It allows permission control over some parts of program and systematic and permanent control of user's actions. To avoid confusion with Logger: this loggs are related on user actions only. Multilanguage working example could be seen at [OB CMS](#) topic.

Some interesting snippets:

- *Some interesting security features from iShares:*
http://www.w-app.net/CV/iShares_second_level.pdf
- *Logg of autobooking engine:*
www.w-app.net/iBookingsAutobookingLogg.txt
- *Framework project has interesting suite for configuration & instantiation of classes:*
www.w-app.net/CNF.png
- *Example of iDTP's logg:*
http://www.w-app.net/Example_Logg.txt
After composite (just a little bit o composite logg is here, then starts thread loop) finished its work, thread engine started with dispatched tasks. Last started thread, in this logg with number 30, has been checked at ordered period for its status. After its status become 0, means its job is over, so engine could try to close it. After that engine tries to check status of other threads, and since they all have started before number 30, they are finished too, so all of them engine attempts to close. This is just a little part, specially shortened for this occasion, because this logg takes care about few hundreds of data, and some of them may have hundreds of logged different states.
Second example:
http://www.w-app.net/Example_Logg2.txt
This logg shows relation between parent and child process. Also it shows execution times of forked children, and order of its closing. It depends of the amount of work every particular thread had to do, so, they was started in sequential order (0,1,2,3,4,5,6,7,...), and finished, as logg shows: 0,7,1,9,8,18,15...
- *Since iDTP is server distributed application, time of production is directly connected with installed hardware. It scales well with added resource, so, if user decide to double speed of production, he has simple solution.*
User decides what part of application will work where, at which server in particular time.
Screenshots:
<http://w-app.net/CV/SDA/>
- *Comparison among few usual formats (XML, JSON...) and some improvements for REST and other kinds of inter-applications communication. Testing application will generate and compare different files. Also some ideas for big data:*
<http://w-app.net/CV/FC/>

iDTP

Please find some articles from Croatian IT magazine "Mreza" describing one of projects I've been working on:

1. www.w-app.net/Mreza_travanj_2008_85.JPG
2. www.w-app.net/Mreza_travanj_2008_86.JPG
3. www.w-app.net/Mreza_travanj_2008_87.JPG

Above is original article in the Croatian language. English translation:

http://www.w-app.net/iDTP_English.doc

iDTP is “one of the google-ed out anything multiserver environment in helps if one of servers by dispatching tasks at more, and each of them current business process

iDTP is a set of strongly rely on the typical materials, such as ads issued by the company (user), practically “alone agree” (prepare themselves) for printing even without the need for human interventions. However, it is not everything - not only iDTP is able to transfer all materials in the form suitable for printing, but the application could generate individual websites with the current content for each of its users.

“Before using iDTP application magazine had volume of 150 pages, in the process of preparing the journal and processing about 2800 images, five people worked six days and four nights which is approximately 400 working hours to complete the process of delivery of ads from the advertiser to the finished material ready for printing house. The current situation, when using iDTP, is that Journal of 208 pages with over 3800 photographs wraps one man in 10 hours.”

Those words of project’s owner means that iDTP effectively change 50 graphics:
“Imagine that you have a magazine of 150 pages (With approximately 2800 photographs), which develops team of five graphics six days and four nights (about 400 hours). Then the market dictate to increase the volume to a minimum of 200 pages, and you simply want to gain greater actuality of ads and to reduce production time to a maximum 10 hours from the time when advertisers have submitted material until ready PDF's for printing house. The solution is "easy" - you can hire 52 graphics (520 man-hours in a given time 10 hours) - and the problem is solved! !”

Packages:

Even Swap

<https://www.phpclasses.org/package/9305-PHP-Implement-decision-methods-using-Even-Swaps.html>

Package won second place at Innovation award competition.

Examples:

<http://evenswap.w-a-f.org/>



September 2015 Number 2

Database visualization:

<http://rg.w-a-f.org/>

Click on some table name and database layout will be shown beginning with that table.

Presented database is iBooking’s early version.

kind”, at least I haven’t similar. It works at which ordered redundancy failed. iDTP does its work chosen server(s), one or then forks as many tasks as needs.

technologies that are Internet, and which allow

practically “alone agree” (prepare themselves) for printing even without the need for human interventions. However, it is not everything - not only iDTP is able to transfer all materials in the form suitable for printing, but the application could generate individual websites with the current content for each of its users.

Projects:

Prolog

I made usable interpreter in C from the scratch. Intention is to use it for NLP, ETL and other project where it may need. Data scraping with prolog has new capabilities. [More about that...](#)

Presentation on demand.

ETL

I have some projects available on demand that use prolog for data extraction. From bunch of text files, using for that purpose developed predicates, extract data in rdf triplets.

Agents based approach: for every task pool generate new agent. Web interface shows what is happening during retrieving pages and data extraction.

Screenshots:

<http://w-app.net/CV/ETL/>

Same project with web based approach:

<http://www.w-app.net/etl/index.php>

It does following:

- a) Drop database and create new one.
- b) Get 88 pages from target site.
- c) Parse them by ordered rules and extract data.
- d) Store data in database.

This is kind of presentation, so every step is visible on screen. Also, post festum, it is possible to see what has happened on server...

Credentials on demand.

I have Bcc in IT:

<http://www.w-app.net/diploma.pdf>

208 ECTS; 4,48 grade point average.

And I have MSc in informatics:

<http://www.w-app.net/d/en/>

132 ECTS, 4,61 grade point average.

Sincerely,

Damir Vukres