

## 1. About me.

I'm experienced full-stack developer and I'm working with .NET since 2002 year. Also I'm fan of cryptocurrency and publish on my blog and other place various recipes and articles about cryptocurrency, for example this post about Huobi

<https://www.vb-net.com/HuobiFreedom/Index.htm>

Also I many times created various trading bot with cryptocurrency, this is example of report of one of my trading bot <https://www.youtube.com/watch?v=thf4Va1yRKQ>

## 2. About shutdown of last my huge cryptocurrency project.

Last 3 year I worked as system architect on project cryptochest.io, this is maybe most interesting cryptocurrency project in my life, but this project was unexpectedly shutdown at January 2023 because project had no enough profitable. But a couple last months we speaking many times about project efficiency and currently I clearly see right way to overcome any troubles with project efficiency.

My way is to avoid rent any external IP address and directly connect User Virtual machine to CloudFlare. I like Cloudflare CDN (even my blog <https://www.vb-net.com/> placed on Cloudflare cache) and know how this service can make my last project profitable, we can absolutely avoid to buy new IP range from Hetzner and only this simple step allow project can profitable, but my way with Cloudflare was refused by Cryptochest.io customer and project was shutdown.

Because this adding need more time and efforts from developers, however customer require profit immediately.

## 3. Earning source of project Cryptochest.io.

Common meaning of Cryptocurrency project allow end-user to create and deploy masternodes for various coin, logic of project like Amazon web panel or Azure web panel. After registered user deployed dockers or masternodes than end-user can receive profit for his investment, for example if we see to this dashboard <https://masternodes.online/> we will understand that about 50% of fees sending to masternode owners, this is income of end-users of cryptochest.io project and source of project profit.

Of course only small parts of users has money to full masternode therefore we need service for staking and sharing masternodes between project end-user.

So, project allow user registration and deploy two type of masternodes – on virtual machines and on docker hub.

## 4. Two different type of masternode hosting.

Prefer coin for project was ETH2, this is my post about preparation ETH2 Virtual machine <https://www.vb-net.com/Eth2/Index.htm>, but of course this decision was a preference for project customer, normal project must support all of 200 coins.

Also we can deploy and start coin as Docker container, I publish a couple of our coins on this page and DockerHub as example

<https://www.vb-net.com/DockerizingMasternodes/Index.htm>, in reality project can working with 200 coins.

What difference between two types of hosting? Price for end-user firstly and user access. Docker masternodes fill coin databases maybe a week (depends of resources), VM databases usually started during 24-48 hours and allow user full control own server with SSH.

## 5. Project database.

Project had huge database, a part of database I have published to Github

<https://github.com/Alex-1557/KvmHostingManagementDbWithTrackingChanging/blob/main/KvmHostingManagement.All.NoData.2022-06-10-18.41.33.sql> - project database worked without transaction, but with full snapshots of users data and tracking changes.

## 6. Various special project future.

Project used a lot of various unique future from special encoder

[https://www.nuget.org/packages/RFC4648\\_Base32\\_Encoder/1.0.0](https://www.nuget.org/packages/RFC4648_Base32_Encoder/1.0.0)

to custom message bus instead RabbitMq or Kafka.

Other unique library I have develop for this project was DockerMonitor - <https://www.vb-net.com/DockerMonitor/Index.htm> in order to tracking docker events and pass docker events to message bus and finally push docker events to Frontend by SignalR -

<https://www.vb-net.com/AngularMosaic4/Index.htm>

## 7. Project hosting.

Project based on Hetzner datacenter and all Virtual machines based on KVM server, I publish a couples articles about increase server performance, for example <https://www.vb-net.com/MapKvmCpu/Index.htm>

This is my Hetzner installation script to initialize servers for this project - <https://www.vb-net.com/HetznerInstallationScript/Index.htm>

## 8. Project team.

Project owner set up strong restriction about programmers salary, therefore nobody working with project more than 2 months, programmers permanently fired and most of important and critical parts I must written by my own hand, for example I have write custom message bus (instead Kafka or RabbitMQ), this is description of my message bus <https://www.vb-net.com/SshQueueNotificationServer/Index.htm>

Project team worked on Private GIT repository I have installed for projects <https://www.vb-net.com/PrivateGIT/Index.htm>

## 9. Backend.

Last Backend working on NET CORE 6, this page contains list of project Backend API -

<https://www.vb-net.com/AngularBackendResult/Index.htm>

And this is my TDD methodology to testing Backend API <https://www.vb-net.com/TDD-Technique/Index.htm>

## 10. Admin panel.

Project has huge admin panel on Blazor, this is historical idea from previous project architect from England, he prefer another engine to start virtual machine (ESXI 7, I start working on this project after Paul under VmWare

<https://www.vb-net.com/VmWare7/Index.htm>) and another datacenter (DigitalOcean), but I (1) transfer project to Angular from Blazor, (2) to Hetzner from DigitalOcean, (3) to KVM from VmWare.

When I worked on Blazor with that project I had publish various Blazor recipes, for example <https://marketplace.visualstudio.com/items?itemName=vb-net-com.BlazorInMemoryDatabaseCRUDexample>

What I did not have time in order to make project profitable - to delete using External white IP for user's masternodes and replace external IP to Cloudflare VPN.

## 11. Frontend.

You can see Frontend of this project on page

<https://www.vb-net.com/AngularMosaic/Index.htm>

On this post I explain notification from Backend to Frontend in my project, unfortunately Frontend on this project has not fully completed. I propose project owner to create Discord bot (I'm author of common discord bot project templates, for example

<https://marketplace.visualstudio.com/items?itemName=vb-net-com.VbNetCoreDiscordBotProjectTemplate>

or create Telegram bot in order to collect more users without full workable Frontend, only with text mode with Discord or restricted graphic mode on Telegram (like this

<https://core.telegram.org/bots/webapps>)

Also I'm very interesting to create exactly mobile application, even when site Frontend not finished, this is my way to create mobile application -

<https://www.vb-net.com/NSPreview/Index.htm>

(special implementation of Angular what allow translate Angular to Apple iPhone device and to Java for Google Android mobile device), but unfortunately this is also was refused by project customer and project finally shutdown at Jan 2023.

## 12. Analyze project mistakes.

Project shutdown because we must permanently buy IP range from Hetzner, when we started and I replace previous architect Paul from GB, one IP had price \$0,8, final price increase 3 times more. Customer refuse to rewrite project to Cloudflare.

In order to connect more end-user I propose made accent to (1) Discord bot on simple text mode, (2) Telegram bot, (3) Shift accent to mobile application. Even with not fill finished Frontend. Customer refuse this proposal and create strong sequence - we must have absolutely workable site Frontend on big monitor and and only than we can start working with mobile apps and bots. This is mistake, of course, we could start all types of Frontend on parallel.

Staff turnover was so huge and we have no time to create high quality Dockers and Vm for all coin, therefore in practice we can not support all coins as expected.

Currently I see all our mistakes and know how to make profitable cryptocurrency projects.