



# Dmitry Inyutin

inyutin.da@gmail.com | 89154336070 |  inyutin |  inyutin

## Education

---

Bachelor in Applied Mathematics and Computer Science  
Moscow Institute of Physics and Technology,  
Department of Innovation and High Technologies.

2016 - 2020

## Work experience

---

### Developer at Yandex

Work as backend developer at the video streaming department.  
Implemented logic to release computing power, which allowed to reuse about 40% of machines.  
Developed API for internal services.

Current

### Assistant at MIPT

Assisted with the “Theory and Practice of Concurrent Computing” course.  
Reviewed students’ homework and clarified concepts from seminars and lectures.

Spring 2019

## Knowledge of programming languages

---

### Python

Work on distributed highloaded service. Wrote small game and some scripts.  
Familiar with asyncio, tornado, pygame and main math libraries.

### C++

Actively use it for educational projects. Familiar with standard containers and synchronization primitives.

Also experienced in Java, JavaScript and Go.

## Participation in hackathons

---

Cryptobazar, Moscow, September - December 2018  
Various applications related to blockchain: pair encryption,  
mobile crypto-wallet, virtual machine for WebAssembly.

Winner

Global Changers 2, Moscow, March 2018  
Web service that creates interactive Customer Journey Maps.

Winner

## Projects

---

### ViBoard

The project that grew out of hackathon Global Changers 1. It was a video service where all the data was stored distributed on the IPFS network. We tried to build a community based economy. Every day a certain number of new coins, which were distributed to the most active users in a day, appeared. We supposed coins cost would to be ensured by the popularity of the platform, but the project didn't took off.

### CASPaxos

CASPaxos is a distributed register without a log.

The main idea of CASPaxos is an attempt to replicate the state of the register, not the log. I wanted to figure out how it works and wrote my little implementation.

### TinyFileSystem

Toy file system model. All the information is stored in one file and access to it is provided through a special application-interface. I have implemented all the basic commands for working with folders and files.