

Senior Full Stack / Backend Engineer

Architecture · High-load · Distributed Systems · Integrations

Contacts:

Email: serg_arx@i.ua

Telegram: https://t.me/serg_arx

Phone: +380995861530

Over 15 years of commercial software development experience. Primary focus on backend and systems engineering, with practical experience across the full stack. Strong hands-on experience with PHP/backend, JavaScript, complex system design, integrations, databases, high-load systems, asynchronous and background processing, and business process automation.

Designed and developed a custom extensible build/compilation framework for web application generation, with modular compilation pipelines, preprocessing, code generation, localization, asset bundling, and pluggable compilation targets.

Experienced in building systems and individual services from scratch, maintaining and evolving legacy projects, and gradually improving existing architecture.

Worked on systems handling large volumes of requests and data, complex integrations, distributed processing of operations, queues, caching, and long-running background processes.

Across different projects, my responsibilities have extended beyond development to architecture design, technical decision-making, complex task decomposition, and production system optimization.

Key Competencies

Designing systems and services from scratch

- * Modular architecture
- * OOP, SOLID, MVC, design patterns
- * Modular design and separation of concerns
- * Monolithic, modular, and microservice architectures
- * REST APIs and integrations
- * High-load systems and performance optimization
- * Asynchronous and background processing
- * Queues and task dispatching
- * Event-driven and message-driven approaches
- * Caching
- * Database design and optimization
- * Complex business processes and their automation
- * Integration with external systems and APIs
- * Legacy systems
- * Debugging and production troubleshooting
- * Logging and monitoring
- * API security, request signing, and cryptography

PHP / Backend

- * PHP — 10+ years of commercial experience
- * Laravel
- * Yii 1 / Yii 2
- * REST APIs
- * Composer
- * CLI
- * Code generators and parsers
- * Smarty, Twig, XSLT
- * cURL

Databases

- * MySQL, PostgreSQL, MS SQL, Oracle
- * Database schema design
- * Normalization and denormalization

- * SQL query optimization
- * EXPLAIN and execution plan analysis
- * Indexing, including covering indexes
- * Views
- * Stored Procedures / Functions / Packages
- * Transactions
- * Triggers
- * Precomputation and caching of complex operations
- * Database optimization for high-load systems
- * Working with large volumes of data

Queues, Asynchronous Processing & Infrastructure

- * RabbitMQ, Beanstalkd
- * Redis
- * Task queues
- * Scheduled/background jobs
- * Task state management
- * Locking mechanisms and prevention of duplicate processing
- * Docker
- * Vagrant
- * Apache, Nginx
- * Git, SVN
- * Dev / Staging / Production environments
- * Google Cloud Platform (GCP)

JavaScript / Frontend

- * Native JavaScript
- * JavaScript + jQuery — 10+ years of practical experience
- * Angular / Angular 2
- * React + Redux
- * WebSockets
- * HTML5
- * CSS3
- * Bootstrap
- * Less / Sass
- * Gulp, Bower
- * Babel
- * UglifyJS, JSHint
- * Dojo, Raphael, jqPlot
- * Regular expressions
- * Browser APIs
- * Working with binary data and AMF
- * Chrome / Firefox / Opera Extensions
- * Chrome Extensions Manifest V2 / V3
- * LocalStorage
- * Content Security Policy
- * Browser security restrictions

Integrations & External Systems

Practical experience designing and implementing integrations between internal systems and external services:

- * REST APIs
- * API authentication
- * Request signing
- * Hashing
- * Secret keys
- * Public and private key cryptography
- * Cryptographic signatures
- * Encryption
- * Payment integrations
- * WebMoney, Interkassa, PayPal
- * FTP / SFTP
- * XML

- * iPaaS
- * Supplier integrations
- * Frontend/backend integration

Other Technologies

- * Java — production experience
- * Tomcat
- * JSP
- * Delphi
- * Oracle / Toad
- * Salesforce Commerce Cloud
- * Controllers / Pipelines
- * WordPress
- * Drupal
- * JIRA

Technical Responsibility & Architecture

- * Independently owning and developing technical areas
- * Requirements analysis and technical solution selection
- * Architecture design
- * Complex task decomposition
- * Code review
- * Working with legacy systems
- * Gradual architectural improvements
- * Identifying bottlenecks
- * Production system optimization
- * Collaboration across frontend, backend, infrastructure, and external services
- * Technical consulting and knowledge sharing

Built fully operational software systems from the ground up, taking projects from initial architecture and core implementation to a working production-ready state

Designed and developed a custom extensible build/compilation framework for web application generation.

The system included:

- a modular architecture with extensible compilation targets and transformation modules;
 - a custom DSL for defining source structure and build configuration;
 - a preprocessing and code-generation pipeline;
 - recursive source-file inclusion and composition;
 - a shared state mechanism between modules during compilation;
 - PHP and JavaScript code generation;
 - a custom localization system and language-resource generation;
 - JavaScript/CSS bundling and optimization;
 - CSS preprocessing with custom SCSS mixins;
 - generation and processing of .htaccess, robots.txt, and other specialized formats;
 - encryption and minification of generated assets;
 - an administrative web interface for controlling the build process;
 - the ability to introduce new compilation targets without modifying the core engine;
- The architecture was designed around extensions and modules, keeping the core compiler independent from specific file formats and transformations. New compilation targets could be introduced as separate extensions while reusing the common compilation pipeline.

High-Load Banking System

Participated in the architectural design of a system handling a large number of external requests and financial operations.

The architecture included:

- * External API bus
- * Initial request processing and protection
- * Load balancing
- * Horizontal scaling
- * RabbitMQ

- * Distributed processing of operations
 - * Dedicated servers for executing operations
 - * Integration with other internal systems
- Also worked on security, reliability, and load distribution.

Modular Online Game Automation Core

Developed a custom modular core based on a scheduler/timer and a set of independent modules.

Each module had its own logic and configuration and implemented a defined interface, allowing functionality to be extended without modifying the core.

Worked with:

- * AMF
- * JavaScript binary data
- * ActionScript 3 structures (Externalized Object, String Reference)
- * Network protocol analysis
- * Traffic testing
- * Virtual machines

The system is used by several thousand users.

Video Transcoding Service

Developed an automated video transcoding service providing:

- * Video upload
- * Format selection
- * Background processing
- * Result storage
- * Retrieval of completed files
- * Storage of processed videos for subsequent playback

Website & Frontend Resource Generators

Developed systems for automatic website generation based on templates and modules.

Implemented:

- * HTML generation
- * JavaScript generation
- * JavaScript file merging and optimization
- * Minification
- * CSS generation
- * Less / Sass processing
- * Base style overrides
- * Responsive frontend resource generation
- * Build-time optimization

Also developed frontend build processes using Gulp/Bower and custom generation logic.

Browser Extensions

Developed Chrome / Firefox / Opera extensions.

Practical experience with:

- * Manifest V2 / V3
- * Content scripts
- * Background scripts
- * iframe communication
- * JavaScript injection
- * LocalStorage
- * CSP
- * Browser security restrictions
- * Dynamic handling of advertising content
- * Automatic logic updates
- * Analytics collection and operational monitoring

Professional Experience

2009 – March 2012

Alongside my studies, worked on maintaining, fixing, and developing e-commerce websites.

Separate technical project — a C++ compiler for an Intel 8080 processor emulator, including:

- * Lexical analysis
- * Syntax analysis
- * Module inclusion
- * Label generation
- * Target code generation
- * Compilation result storage

Banking Sector

March 2012 – April 2013

Delphi + Oracle

- * Oracle / Toad
- * Functions / Procedures / Packages
- * Transactions
- * Triggers
- * Complex financial calculations
- * Working with large volumes of data
- * Precomputation of complex operations
- * Storing calculated results for reuse

Radical Computing

May 2013 – November 2014

- * Maintenance of existing and development of new projects
- * Data collection and processing systems
- * Metrics
- * MVC
- * Drupal
- * Drupal hooks
- * Modular systems

WebSunGroups / Kadam

November 2014 – November 2015

Worked on advertising and RTB systems:

- * APIs
- * Redis
- * JavaScript
- * Code generators
- * JavaScript injection
- * Frontend/backend integration
- * Large number of independent advertising modules
- * High-load systems with large request volumes

White Rhino Marketing

April 2016 – May 2017

- * Startup development and maintenance
- * Symfony backend
- * Frontend
- * APIs
- * Landing page generation
- * Gulp
- * Asynchronous generation
- * WebSockets
- * Charts

- * Client-side libraries

Marketing4u

May 2017 – September 2017

- * PHP / MVC
- * JavaScript
- * Angular 2
- * Frontend build tools
- * Startup development and maintenance

Alfa-Bank

October 2017 – October 2018

Participated in the architectural design of a system for working with external APIs and financial operations.

Worked with:

- * API bus
- * Load balancing
- * RabbitMQ
- * Distributed processing
- * Integrations
- * Security systems
- * System scripts requiring elevated privileges

PosMasterUS

November 2018 – April 2019

Development and maintenance of systems for retail stores in the US.

- * Java
- * Calculations
- * Data processing
- * Tables
- * Charts

Astound Commerce

May 2019 – May 2020

Development and maintenance of e-commerce systems based on Salesforce Commerce Cloud.

- * Controllers
- * Pipelines
- * E-commerce objects
- * Development and maintenance of existing stores

Futurra Group

May 2020 – March 2021

- * Maintenance of existing systems
- * Integrations
- * Non-standard approaches to data storage and retrieval
- * Existing infrastructure maintenance
- * Optimization of individual components

Autodoc

April 2021 – August 2026

Development and maintenance of complex e-commerce business processes and integrations with a large number of suppliers.

Main areas

- * Supplier integrations via API, FTP/SFTP, and iPaaS
- * DROP / JIT / PRE-Purchase
- * Automatic switching between purchasing models
- * Supplier onboarding and maintenance
- * External APIs

- * FTP / SFTP
- * XML
- * Google Cloud
- * Processing large supplier files in a cloud environment
- * Background processing
- * Scheduled jobs
- * Resource optimization
- * PHP / Python
- * DTOs for structured data transfer between components and services
- * Logging
- * Monitoring
- * Dedicated services for supplier operations
- * Supplier dashboard / portal
- * Complex automatic purchasing logic and fallback between suppliers

Worked on an automated product purchasing system integrated with a large number of suppliers. Once a purchasing process started, the system contacted suppliers sequentially, starting with the most advantageous offer. If a purchase attempt failed, the system automatically proceeded to the next supplier.

After a successful purchase:

- * The supplier order ID was stored
- * The internal order status was updated
- * The subsequent tracking process was started
- * Statuses were automatically checked by background processes

For large supplier files, part of the processing was moved to Google Cloud, reducing the load on the main application servers.

For scheduled processing, locking mechanisms were used to prevent a new processing job for a specific supplier from starting until the previous one had finished.