ALUPOAIE ALECSANDRU

Senior Software Developer

Personal Data

☎ +40747887752
✉ alupoaie.alecsandru@gmail.com
S skype: alupoaie.alecsandru
Location: Cluj, Romania
in LinkedIn:
https://www.linkedin.com/in/alecsandru-alupoaie

Skills

C# .Net
Api Services
Web Applications
Cloud Architecture
Full Stack Developer
Environments Setup

Clean Coding
Software Design
Application Testing
Flexible Team Player
Requirements Analyst
Team Leader

Software Developer

Passion
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
C#
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□....
Javascript
□□□□□□□□□□□□□□□□□□□□□□□□□□□...........
SQL
□□□□□□□□□□□□□□□□□□□□□□□□□□□...........
Continuous Delivery
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□..
Communication
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□......

EDUCATION

Master’s degree in Computer Engineering Science
Bachelor’s degree in Computers and Software Engineering

WORK EXPERIENCE

Nordcloud, contractor **2021 Dec - Present**
- Tech lead and team leader
- Deliver Cloud Infrastructure, automate deployments
- Find cloud connectivity solutions maintaining security
- Design components integration
- Technical Proof of Concepts on migrating technologies
- Cloud Infrastructure as code, environments creation

Evozon in Cluj, RO 2015 Jul - 2021 Dec
- Tech Lead Full Stack Software Developer
- Responsible of creating new applications
- Discussions with the client in solving architectural challenges
- Analyze data flows to improve our apps
- Continuous Integration and Continuous Deployments
- Solving technical challenges across teams

Freelancer in Suceava, RO 2013 Jan - 2015 Jul
- Developed Full Stack Solutions
- Took new challenges and always learning
- Communication skills in talking with the clients
- Had the opportunity to work with multiple programming languages
- NodeJs APIs integrated with AWS resources: S3, DynamoDb

HIGHLIGHTS

- Flexible in my domain of work, front-end and back-end
- Continuous Integration continuous delivery solutions
- Azure Services (functions, storage, app configs, service bus...)
- Amazon Web Service (lambda, s3, sqs, sns, api, dynamoDb...)
- Agile Methodologies of working
- Front-end experience with React, Vue and Angular
- RabbitMQ to handle complex data flows
- Automation Testing along with Continuous Integration
- Azure Cloud integration, cover security
- Network infra configuration, firewall rules, virtual networks
- GraphQL for handling complex custom endpoints
- Layerd Architecture, Microservices, Design Patterns
- Managing distributed systems
- Research various architectures lead by POC implementations
- Implemented alerting and monitoring solutions with slack alerts
- Always getting involved in discussions with the client

Languages

Romanian

● ● ● ● ●

English

● ● ● ● ○

Side projects and jobs:

Java Desktop Applications:
- show graphic analysis on physical displays;

Windows Forms Desktop Applications:
- develop a software for a local gym

NodeJs Api Rest WebServices:
- chat bots with facebook integration
- api endpoints

Websocket communication:
- control my pc from my phone

Interests:

Self improvement
Keeping up to date with the technologies
Being involved in the development process
Learning by helping
Enhancing knowledge in Docker and Kubernetes

Activities and Hobbies:

Snowboarding
Drawing
Dancing

WHAT DID I REALLY DO LATELY?

I've been working as a **back-end developer** with **Azure** technologies. A high overview of the components would be represented by **storage accounts** that get the payloads as **blob files** in different formats. **Azure functions** have triggers that take and process the inputs, doing some validations and depending on the result the output is either a **cosmos db** record or a **service bus** payload placed on a queue. **Application Insights** is being used across solutions to log and monitor dataflows and exceptions. For sharing app settings across different functions we used azure app config that also has integration with **keyvaults** in order to store sensitive data.

All the resources are running under a **vnet** where the subnets are being managed separately and provided to the deployment scripts. The infrastructure is being delivered using **azure bicep** declarative syntaxes.

The **environment costs** are different so this is an important factor when deploying the environment as this will affect also features of the components. Keep development environments with **low costs** and production environments with **high performance configuration**.

The business logic was mostly in azure implemented with **durable functions** and the security around it was delivered using **role-based access control**. Deploy Automation Accounts, azure resources and configure **Pipelines** for **CICD** using **Azure Devops** and **Github Actions**. Different integrations with SQL Cloud Databases were required, either **PaaS** or **SaaS** and each of them had different authentication systems, password rotation mechanisms.

As repositories most of the projects were in **GitHub**. The pipeline could automatically download the desired repository in order to do the builds and releases it to different environments. The configurations for of the applications are kept in **azure app configs** where each application requires access and priviledges in order to access the values. The pipeline optimized for **CICD** was developed to handle the entire process, from the **pull requests** stage, to run the **tests**, create the **build** and **release** to all environments. Service Connections where used from Azure Devops to access resources in a maximum security manear.

Besides the Azure technologies that I used lately, in my previous projects I had to upgrade projects from .NET Framework to **.NET Core** and **.NET 5** and **6**. I worked with **RabbitMQ** as alternative to **Azure Service Bus**. Some of the front-end applications required **GraphQL** endpoints for better flexibility. Also on the front-end I built applications using most of the popular frameworks like **Angular**, **React** and **VueJs**.

Before the current project that was in Azure I worked integrating a series of **microservices** in **AWS**. The resources used there were simillar.

- ## Strong background in these as well:

- Leading the internship 2 years in a row on different technologies (**.net full stack**, **javascript**). With the asp team we used **React** with **web APIs**, with the javascript team we created a **framework from scratch using only javascript**.
- Built **NodeJs** APIs and interaction platforms.
- **Selenium** for Testing Automation
- Implemented different **site crawlers**, for youtube in order to download references for different playlists or scan prices for products and alert when the price drops.
- Deliver cloud infra using **Terraform** on Azure and AWS.

Lets have a technical overview

- **Azure DevOps:**

Maintain Builds And Releases Process
Migrate classic releases to yml Pipelines
CICD Automation
GitHub Actions
SQL Credentials Randomization
Bicep Scripting

Optimize pipelines for parallel deployments
Setup project permissions
SonarCloud integration for CICD
Terraform
IaC (infrastructure as code)
PR Process Setup

- **Azure:**

Azure Functions [classic, durable functions]
Service Bus [queues, topics]
Application Insights
App Service Plans
Azure Sql Databases
VM Configurations
Azure App Configs
RBAC Permissions Automation
Storage Accounts

VNet Configuration
Private Endpoints setup
VNet Peering, Point 2 Site Configuration
Cosmos Db partitions configuration
Subnet Configurations
Firewal Rules Setup
DNS Configuration
KeyVaults
Managed Identities

- **AWS:**

Lambda Functions
S3 storage provider
DynamoDb
GitHub Actions
AWS SNS
AWS Lightsail
AWS IAM

EC2 Instances
DNS Configurations
Terraform
AWS Beanstalk
AWS SQS
AWS Cloud Formation
AWS Management Console

- **Coding:**

Dependency Injection
Api REST Patterns
.Net with C#
Reusable patterns

Monolithic architecture
Microservice architecture
Batching strategies
Authentication, Authorization