

Software / Hardware developer

Where “not possible” is no option, where performance and innovation really matter, where you achieve what others couldn't, where Cloud meets Mobile and Embedded – that's the place to be for me.

1 Overview

I'm a passionate software/hardware developer with many years of experience.

The four major areas of my programming expertise are: web-based, mobile, embedded and desktop programming. There is also a fifth area: in-depth knowledge of electronics and some experience with developing smart hardware.

I love to push mobile devices to the limits by means of unusual Native apps and by taking non-orthodox approaches to achieve amazing media streaming performance.

More recently I enjoy creating Progressive Web applications (PWAs) where WebRTC is combined with WebGL and TensorFlow (ML).

Not to forget, I also like to dive deep into the Embedded Linux to continue where others stop to get the most out of wireless connections.

2 Keywords

2.1 General

State-of-the-Art, Innovation, Performance, Perseverance, Automotive, Mobile, ML/DL, R&D, OOP, UML

2.2 Programming

- HTML5, CSS3, Javascript, PHP, C++17/STL, SQL, C#, Python
- Android/Native (Kotlin/Java/JNI), iOS (Swift/Objective-C)
- Node.js, React.JS, WebRTC, Media streaming, Image processing
- Linux (User space/kernel), OpenWRT, mac80211
- OpenCV, Neural networks, TensorFlow, Keras, WebGL (THREEJS)
- .NET, WPF, Silverlight, DirectShow

2.3 Protocols

HTTP, TCP/IP, IEEE 802.11, MJPEG, H.264, V2X, ISO 17215, CAN, UDS, KWP2000, ASAM

2.4 Operating Systems

Linux (Desktop/Embedded/OpenWRT), Android, iOS, Windows

2.5 Hardware

Digital, Analog, Wireless, ARM, MIPS, Complete Design

3 Work Experience (summary)

3.1 Currently

From 2015 to now: working from Sioux/Logena for a customer (Orlaco) as Mobile, Embedded and Web Application developer on a high end, low latency automotive video streaming and high throughput wireless routing system. The programming languages, protocols and technologies: Android, iOS, Linux/OpenWRT, Media streaming, Kotlin, Swift, C++17/STL, HTML5/CSS3/Javascript/PHP, IEEE 802.11, ISO 17215, V2X etc.

From 2016 to now: Web-based Video conferencing and game system (personal project). The programming languages, protocols and technologies: HTML5/CSS3/Javascript/JSX, React.JS, Node.js, WebRTC, Linux.

From 2019 to now: Mobile/Web-based Machine Learning, Image recognition and 3D modelling system (personal project). The programming languages, protocols and technologies: Kotlin, HTML5/CSS3/Javascript, Node.js, Python, TensorFlow/Keras, OpenCV, WebRTC, PWA, WebGL (THREEJS).

3.2 In the past

From 2013 to 2016: Software/Hardware development of a Cloud Vehicle Diagnostic system at Logena Automotive. The programming languages, protocols and technologies: C++11/STL, OpenWRT, HTML5/CSS3/Javascript/PHP/MySQL.

From 2014 to 2014: Software development of a customizable tool for smart file conversion containing structured data at Logena Automotive. The programming languages, protocols and technologies: C#/WPF, XML, MATLAB.

From 2010 to 2016: Software development of an Automotive Windows Diagnostic Tool at Logena Automotive. The programming languages, protocols and technologies: C#/WPF, KWP2000, ASAM.

From 2008 to 2010: Desktop and Web Application developer at MASC B.V.. Used programming languages: C# / ASP.NET / T-SQL / Silverlight.

From 2001 to 2008: Working as software developer at Systec Designs in Amsterdam. Tasks: developing native Windows applications for controlling and monitoring the hardware developed by this company (e.g. monitoring and controlling card readers, radiographic systems, light engineering etc.).

From 1999 to 2001: Running my own Electronics Repair company (Netko Electronica Reparaties) in Meppel.

4 Skills (summary)

4.1 Software Development:

- Android/Native (Kotlin/Java/C++) applications
- iOS (Swift/Objective-C) applications
- Smart Web applications (HTML5/CSS3/Javascript)
- Innovative web technologies: PWA, React.JS, WebRTC, WebGL, Google Maps API
- Server-side programming: Node.js, PHP, MySQL
- Linux applications (C/C++ 17/STL)
- Linux drivers (net/usb devices)
- Windows applications (C#/WPF/C++)
- Media streaming (H.264, MJPEG, Android/iOS hardware accelerated decoding, GStreamer)
- Machine Learning (TensorFlow, Keras, Python)
- Automotive embedded software development (C/C++ 17/STL)
- Automotive protocols and standards: CAN, UDS, KWP2000, ASAM, DTCs, SOME/IP, V2X etc.
- Complete architecture of automotive embedded cloud systems for global data exchange and remote diagnostics
- Operating systems: Android, iOS, Linux, OpenWRT, Windows, Bare Metal systems

4.2 Hardware development:

- Full development of High Performance, 4G/Wi-Fi/Ethernet enabled (automotive) embedded devices
- Excellent knowledge of electronics

4.3 General:

- Object-oriented programming
- UML
- Scrum

5 Education

A foreign Bachelor's degree (Electrical and Electronics Engineering) at Electro Technical College Banja Luka (ex-Yugoslavia; 1985-1989). I am a true autodidact loving proven science and technology.

6 Personal information

Name: N. Cvijetić

Nickname: Nedjo

Date of birth: July, 4th 1971

Nationality: Dutch

Phone numbers:

+31-485-750728 (after 18:00 h)

+31-6-55590902 (mobile)

Email: job@nedjosoft.com

Website: www.nedjosoft.com

Languages: Dutch, English, Serbian/Croatian

7 Work Experience (detailed)

7.1 Currently

7.1.1 From 2015 to now: Working from Sioux/Logena for a customer (Orlaco) as Mobile, Embedded and Web Application developer.

Project:

Complete Automotive low-latency Video streaming system

Description:

Wireless Video streaming from blind-spot cameras onto Android/iOS or custom HMI devices. The system consists of several types of Orlaco cameras, multiple mutually paired custom Wi-Fi devices and a display device (either an Android/iOS mobile device or a custom Display/HMI device). Some applications are: Automotive Cabin-Trailer wireless Video connection, Crane Video systems, etc.

Challenges:

- Very high data throughput
- Video low-latency streaming on Android/iOS devices
- Reliable Wi-Fi connections even in congested areas
- Simple management of multiple Wi-Fi units

Tasks:

Complete software design and development of all system components including Embedded, Mobile and Web software.

Languages and techniques:

- Environment/Operating systems: Android/iOS/Linux (Embedded; applications and drivers)
- Programming languages: C/C++17, Kotlin, Swift, Java, Objective-C, HTML5/CSS3/Javascript, PHP
- Suites and protocols: OpenWRT, Linux mac80211, Android/iOS low-level hardware-accelerated video decoding, ISO17215 (SOME/IP), IEEE 802.11 (WLAN), V2X (experimental)

7.1.2 From 2016 to now: Web-based Video conferencing and game system

Project (personal):

Special type of Web Audio/Video conferencing and online gaming.

Description:

A fully web-based conferencing system initially designed for remotely attending the meetings in the Kingdom Halls (Jehovah's Witnesses) with the possibility to easily participate.

Due to the COVID-19 crisis, another part has recently been added: Live Video chats with online games (to ease the loneliness).

Challenges:

- Very simple User Interface so that older people can use it too
- Reliability (even with poor Internet/Wi-Fi connections)
- Supporting various types of devices and web browsers

Tasks:

Complete software design and development of all system components.

Languages and techniques:

- Environment/Operating systems: Web applications, Linux (WebRTC SFU software)
- Programming languages: HTML5/CSS3/Javascript, React.JS, Node.js, MySQL
- Suites and protocols: PWA (Progressive Web apps), WebRTC, Media streaming

Links:

- Kingdom Hall Link (credentials required): <https://www.polymeeet.org/>
- Video chat + online gaming (PIN required): <https://www.polymeeet.org/games/>

7.1.3 From 2019 to now: Mobile/Web-based Machine Learning, Image recognition and 3D modelling system

Project (personal):

Detecting objects, human poses and gestures in live Video streams, animating 3D characters based on human poses and applying it in online game systems.

Description:

One part of the project is parsing live camera streams in order to detect objects or human poses and gestures either as an Android native app or a PWA (Progressive Web app) using machine learning. The project also involves creating and training own models.

Another part of the project covers creating and animating 3D characters based on the detected data (PWA only).

The project also includes an online game system with the Audio/Video Web conferencing.

Challenges:

- Reliable detection of objects, poses and gestures
- Collecting large amount of data for training of the models
- Low-latency online interactions

Tasks:

Complete software design and development of all system components.

Languages and techniques:

- Environment/Operating systems: Android, Windows, Web applications
- Programming languages: Kotlin, HTML5/CSS3/Javascript, Node.js, Python
- Suites and protocols: TensorFlow/Keras, OpenCV, WebRTC, PWA, WebGL (THREEJS)

Links:

- Game (under development): <https://www.polymeet.org/moves/>

7.1.4 From 1999 to now: Various smaller personal projects

Projects:

Various projects on different areas (Mobile, Web, Embedded and Desktop apps)

One of these smaller projects was a high-performance Google Maps web application for drawing large amount of map markers and area polygons with hundreds of points

7.2 At Logena Automotive (from 2010 to 2015)

7.2.1 Summary

Developing Automotive Desktop Tools, Embedded programming (Tools and ECUs). Also charged with the architecture and development of a global, Cloud enabled embedded system involving sophisticated hardware development too.

7.2.2 Some projects

- Octolink – a global vehicle communication system involving several development areas: embedded, cloud (client-side and server-side) and smart hardware.
 - *Product type*: Cloud system
 - *Tasks*: system + software architecture, main software development, hardware architecture + design
 - *Languages and techniques*: C++ 11/STL, HTML5/CSS3/Javascript, PHP/MySQL, embedded Linux, OpenWRT, TCP/IP, HTTP, SPI, UDS, CAN, KWP2000, ASAM etc.
- Smart Translator – An adaptable tool for smart file conversion containing structured data (like XML, JSON, MATLAB, various proprietary project file formats etc).
 - *Product type*: Windows Desktop application
 - *Tasks*: Software architecture and development
 - *Languages and techniques*: .NET/C#/WPF
- Automotive Diagnostic Tool – A Windows application written in .NET/C#/WPF used for automotive diagnostics, secured with hard-lock keys
 - *Product type*: Windows Desktop application
 - *Tasks*: Development/maintenance
 - *Languages and techniques*: .NET/C#/WPF
- Various Desktop and Cloud applications using .NET/C#/WPF, HTML5/CSS3/Javascript, PHP/MySQL

7.3 At MASC B.V. (2008-2010)

7.3.1 Summary

Mainly engaged in developing various types of Desktop / Web applications.

7.3.2 Some projects

- Visualizing and mapping of thoughts and concepts
 - *Product type*: Desktop / Web applications
 - *Tasks*: development
 - *Languages and techniques*: C#, ASP.NET, Javascript, WCF, T-SQL 2005, IIS-6, .NET 3.5, VS-2008

- Conducting extensive automated text analysis and administration of results
 - *Product type*: Web Application / Desktop Application / Windows Service application
 - *Tasks*: partial software architecture and development
 - *Languages and techniques*: C#, C++, ASP.NET, Javascript, T-SQL 2005/2008, Windows Services, COM, .NET 3.5, VS-2008

- Silverlight camping data management application involving offline/online synchronization
 - *Product type*: Web-/Out-of-browser Silverlight application
 - *Tasks*: partial software architecture and development
 - *Languages and techniques*: C#, XAML, ASP.NET, Silverlight 3, WCF, .NET 3.5, VS-2008/2010

- High performance media player integrating a DirectShow / Media Foundation module inside a WPF application
 - *Product type*: WPF / Silverlight application
 - *Tasks*: Software architecture and development
 - *Languages and techniques*: C#, XAML, C++, COM, Silverlight 4, WPF, DirectShow, Microsoft Media Foundation, .NET 4, VS-2010

7.4 At Systec Designs (2001-2008)

7.4.1 Summary

Developing Windows native software to support and/or control the specific hardware designed by this company.

7.4.2 Some projects:

- Control of 'intelligent' lighting tiles; visual editing and composition of complex lighting effects; wireless server / client system with different backup systems to ensure continuity, synchronization with the audio streams (e.g. lighting effects to the beat of music), triggering light effects by external sensors
 - *Product type:* Windows native application
 - *Tasks:* Software architecture and development
 - *Languages and techniques:* C++, COM, Windows Driver Model, MAPI, GSM modem control (SMS), FTP, DSP, FFT calculations

- Card authentication system: school locker system with a management module
 - *Product type:* Windows native application
 - *Tasks:* Software architecture and development
 - *Languages and techniques:* C++, Windows Driver Model

- Control of LED signaling boards, Fingerprint authentication systems, measurement equipment etc.
 - *Product type:* Windows native applications
 - *Tasks:* Software architecture and development
 - *Languages and techniques:* C++, Windows Driver Model

7.5 More information

Please contact me for more details about my work experience.