Daniel Johansson

Curriculum Vitae

Career

2018–2020 **System Developer**, *Axis Communcations*

Mainly I have been working with Perl and system integration. This was in a project that aimed to upgrade the old system used by the customer support staff. It consisted of newly bought 3rd party software and both new and old internal systems. Other minor tasks:
- Some system maintenance of our servers together with the team.
- Worked to have proper documentation of the software developed setup and continually expanded.
- Automated minor tasks and tried to automate GDPR requests but wasn't fully completed.
- Collaborated a bit with a neighboring team in machine learning.
- Some frontend. Did not enjoy it very much.

2020–2023 **System Developer, Consultant**, *Lsjbot/Bokus*

Assignment was as a system developer at Bokus. I was working with their legacy Perl code base, adding a few features, and maintaining current features.
- Implemented tools used by customer support and automated processes.
- Implemented a Perl-GCP connection for obtaining data to create feed.
- Created feeds to generate Google Ads via a third party.
- Worked with company-made template language, a mix of sql, html, perl, and template specific key words.
- Upgraded servers:
  - packages for the OS
  - packages for apache
  - Perl 5.18 -> 5.34 (configuring and building)
  - Updating deprecated code
  - Updating installation scripts
  - Debugging odd behaviours when different packages have different versions
- Built an excel-sheet parser for automation of unit tests.

2023–2024 **Machine Learning & Data Scientist Architect**, *Consupedia*

Mainly system development, system administration and dev ops.

○ Development in PHP and Laravel.
○ Dug in legacy code to gain understanding and bridge knowledge gaps.
○ Created module to obtain data about an object in a generilzed, table agnostic (to an extent) way with a generalized way of handling special cases. This was done by mapping simple names of parameters to more complex expressions that are then interpreted in a general way. The complex expression were interpreted as by regular expression to determine table, column and potential post calculation (or special case calculation). The groups captured by the regular expression was then formed into new code that was evaluated in runtime to minimize amount of code to write.
○ System administration on AWS linux server. Create users, set permissions, set ownership.
○ Devops. Setup server, new AWS instance, clone code, setup apache, setup certificate, setup DNS.
○ Microservice in Python calculating Nutriscore.
○ Setup a laravel service for handling microservices in AWS gateway, such as above mentioned microservice.
○ Setup unit tests for said service.

Education

2013–2018 **Master Computer Science**, *Linköpings University*

2010–2013 **Computer Game Development – Programming**, *University Skövde*

2010–2010 **C++ Basic Course**, *University of Gotland*
distance

2009–2010 **Game Development and Programming**, *Malmö University*

2006–2009 **International Baccalaureate**, *Per Brahe High School, Jönköping*

Master Thesis

Title Deep Learning with SkePU - A Skeleton Programming approach

Description Deep learning using convolutional neural networks parallelized with the help of skeleton algorithms provided by the framework SkePU. This thesis is not fully completed.

Bachelor Thesis

Titel	To Grow a City
Description	City generation with help of autonomous agents.

Languages

Svenska	First Language
English	Fluent
Español	Understand the language pretty well as it is my mother's first language

Computer Related

Languages	o C++
	o C
	o Perl
	o Bash
	o MySQL
	o Python
	o Guile
	o Lisp
	o C#
	o Assembler
	o PHP
	o Prolog
	o Java
	o HTML
	o CSS
	o XPath
	o Javascript
Projects	o Grid Game Engine
	o Home made task bar.
	o home_server
	o sc4_region_generator
	o Libgrid
	o trump_tweeter
	o LiU Game Jam
	o Block Frog
	o Lonily Harts.
	o A myriad om smaller script automating processes.
	o See github for more.

Software	○ Linux
	○ Vim
	○ Regular expression
	○ Valgrind
	○ gdb
	○ SkePU
	○ Apache
	○ Ansible
	○ i3
	○ awk
	○ Jira
	○ Confluence
	○ GCP
	○ Trello
	○ AWS
	○ Laravel
	○ Dbeaver

Techniques	○ Scrum
	○ kanban-ish

Linux Specific

Linux   Used since I was 25 or so. All my devices run in it one form or another

Distributions	○ Debian (stable, testing and unstable)

- Borked and unborked again numerous times.
- Fixed issues with graphics drivers, bluetooth, kernel panics, crashing harddrive (/var lost, package manager lost its db) and a myrad of other things.
- Read up on how things work, like bluetooth, package manager, xorg in order learn how fix the issues.
- Wrote my own task bar.

○ Ubuntu

○ Raspbian

- Home server.
- Emulator of old game systems for siblings.
- Assisted friend to setup and troubleshoot the same as previous. point

○ Gentoo (compiled and optimized for one of my laptops)

○ Arch

○ Fedora

○ Amazon-Linux

Projects Specifics

GGE	Grid Game Engine is a game engine for running games that use grids of some kind. I have been working on it since the start of the pandemic. It is written i C++, with some internal C wrappers for handling communication with Guile, which games for GGE is written in. `https://github.com/Vokda/gge`
Danielian NameGenerator	An application that generates names. The names may not be fitting for real people but for a fictional character it would work. Names are generated based on how often letters occur. Occurance of each letter can be changed in the app. This was first a part of a game but the name generation part became much more interesting so it became its own application. `https://play.google.com/store/apps/details?id=danielian.nameGenerator`
Pick Me!	The application handles lists and can randomly pick an item from the list. Its primary application is for teachers. E.g.: A teacher can add all pupils into the list and then randomly select one pupile to answer a question. `https://play.google.com/store/apps/details?id=com.pick_me.daniel.pick_me`
Block Frog	A cross-platform platform game written in C++ with our own network module based on SFML's Network library. There is a button to force synchronization with the host as refernce. Runs on Windows and Mac (it was written before I switched to Linux). `https://github.com/carleryd/Block-Frog`
My Homepage	It's my homepage! Written from scratch, and runs on my own raspberry pi server. You can find it here: `www.devdaniel.eu`

Miscellaneous

Driving Lincense	B