ALEX LUNDIN
Design Engineer
Wattstopper, Legrand
2240 Campbell Creek Blvd
Plano TX, 75074

TELEPHONE (469) 394-5175
EMAIL   alexander.m.lundin@gmail.com

January 21, 2019

Your job posting for caught my attention as I have the necessary programming skills to produce work immediately. I'm electrified about the possibility of being part of your team. I'm a self-starter, I taught myself AutoCAD LISP to become as efficient as possible for my current employer. I'm familiar with creating a system around a desired workflow, I built Legrand's AutoCAD process from the ground up. Title blocks all the way to expert level automation, like custom data extractions for drawing generation later.

Since I am self-taught in the workplace, I think all the above these skills will transfer to any language that is needed in the future. At the University so far, I have studied Java, C++, C, Python and RStudio. I am also familiar with Automated Testing strategies and frameworks. These include Selenium and Junit.

Each semester I applied what I learned to the work force. Auto LISP is missing a regression test suite, so I also wrote my own, similar to Selenium.

I will receive my Bachelor's degree in Software Engineering by May of 2019.

I have 4 years professional experience where I developed code in the work force. I gained many soft skills like collaboration, communication both written and verbal as well as working under pressure.

I hope to hear from you soon because I think this would be a win win for both parties. Please feel free to contact me for a follow up interview at 469-394-5175 or email at alexander.m.lundin@gmail.com.
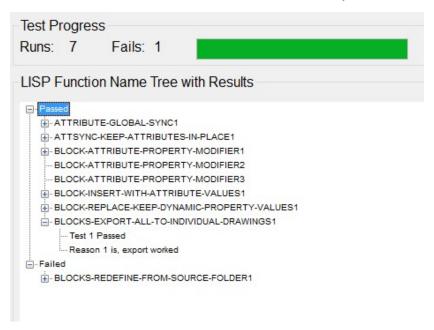
Sincerely,

Alex Lundin

Video demonstration attached on next page.

Regression Test Suite

https://autode.sk/2sByZYq

I learned about regression tests at the University of Texas at Dallas. However, AutoLISP is the language I currently work in, and it does not have an automated testing suite. So, I built my own.

- o   Continuous integration and delivery
- o   Automatically runs regression tests, to ensure the LISP build is successful
- o   I embedded 1 failure, to show how tests appear, for either scenario, pass or fail
- o   Passes and fails are separated cleanly
- o   Helps in reporting on test coverage
- o   Ensures a standard for all releases of my LISP code

Control bar, built with open dcl

https://autode.sk/2SWDVCJ

This adds to the usability of my source code, from the User side, being able to see all commands, in one place.

- o   Provide easy access to entire LISP glossary
- o   Reads LISP source code, and displays user information in the text box, about the command
- o   8 tabs
- o   18 buttons, that dynamically change on tab selection