Hello,

As a professional software engineer, I bring a broad set of skills and experience from working on diverse projects in the automotive, medical and financial industries.  I have utilized many technologies, including C#, WPF, LINQ, DDD, MVVM, SQL, HTML, CSS, JavaScript, Java, C/C++, Perl, XML, XSL, and others.  My software controls the heavy machinery you see in construction areas, medical devices used to save and improve lives and financial calculations that evaluate the risk on investments.

Caterpillar was my first job post-graduation.  I worked within the Core Software group developing hardware abstraction layer (HAL) libraries that directly controlled the circuits to read/set I/O on the new control computers and provided a hardware independent API to the application layers. Our group worked closely with the Core Hardware team to assist with debugging circuit designs.  We also assisted the application developers in the usage of the HAL APIs.  My largest project was the design, development, and documentation of the end of line test suite.  It verified that all circuits functioned on each electronic control module exiting the production line.  This project took me to Mexico where I helped setup up the new production line to use my test suite.  Caterpillar is the acme of engineering companies and made an indelible impression on how to correctly engineer products.

Aksys introduced me into the humanity of engineering by providing the opportunity to help create an in-home hemodialysis machine that resurrects people from a debilitating life of kidney failure.  I am proud to have developed the scheduler subsystem, which keeps track of the patient’s treatment, lab test schedule, the disinfection schedule, and the age of the dialysate.  It executes the necessary sequence of tasks at the proper time to prepare the machine for treatment based on these schedules.  I also assisted in the verification and validation of the device prior to seeking FDA approval.  For my last project, the key design engineer and I were creating an intelligent diagnostic system to test the machine to determine the cause of an error before alerting service personnel via email.

Siemens Medical was a short contract position to update the program which tests the image processing system in their new angiography machine.  The program is used in the factory to test systems prior to shipping and in the field to determine faults, therefore ensuring that the machines are safe for patient use.

Caterpillar brought me back to work in their Software Center of Excellence Wheel Loader team.  On this team, I used the HAL (hardware abstraction layer) libraries I created previously to write a traction control module which analyzed wheel rotational speed and the articulation angle of the wheel loader to determine which wheels to send torque to thus minimizing wheel slippage.  My other responsibilities included being the ClearCase release librarian and tool maker.  I created Make scripts, merged and released new versions of the machine software library to the factory.  My most advanced tool hooked into the pOSEK operating system to analyze the performance of system tasks and software developer defined code blocks in real-time.  Real-time analytics included process run times, function call counters, and a timeline showing what processes ran prior to a system reset.  This allowed engineers to measure the performance impact their code had on the CPU, monitor the CPU load during various tasks, and show the cause of resets due to CPU overload.

Wachovia was fun with a variety of tasks and computer languages I had no previous experience with.  Hired as a C++ and Perl developer my first project was debugging SQL scripts and analyzing data for errors.  For my next task, I updated a suite of financial reports from Crystal Reports version 6 to version 10 and rewrote the highly convoluted VB logic.  Once the reports were updated, I created a multi-threaded bot in Perl to generate over 3000 daily reports.  The bots ran on several servers and coordinated their work via a shared database table.  The bots contained failsafe logic which monitored and recovered from network and database failures, ensuring all reports were completed every morning.  A command line and HTML user interface displayed the status of the report generation process.  For my next project, I designed and developed a user interface to a dashboard system using C#, ASP and AJAX.  This system gave data owners insight into the current and historical status of all their monitored applications, data feeds and processes on the main page.  The ability to drill down provided additional details on sub-processes revealing where failures or delays occurred.  My last project was rewriting an unstable risk calculation program written in Java and 5 other languages.  Cleanup included trimming the application out of a much larger dead application, removing J2EE and Web Logic, developing a database connection pool, properly handling exceptions, fixing logging, and re-writing non-Java pieces as Java.  When finished, this system calculated the risk on $1.5B in investments daily without issues.

JPMorgan was a challenge in that I was hired for a Dev Ops role deploying software to Window servers which was nothing like the Unix servers I was familiar with.  Soon after I joined, another Unix admin joined the team and together we recognized and tackled the lack of organization and processes the current manager and team lead had failed to implement.  I rewrote the deployment scripts written in NAnt such that they performed preliminary tests such as password validation prior to deployment and also gracefully recovered from many exceptions such as network latency and unresponsive systems.  My teammate created processes that required developers to follow proper testing and code promotion procedures that prevented last minute changes prior to a production deployment.  After a year, we had changed the deployment process from being one which started on Friday night and ran non-stop thru Sunday afternoon, with about 90 people on staff rotating on shifts and sleeping in hotels reserved by JPMorgan, to our most complex deployment, which again started Friday night but ended Saturday at noon with only 30 people, half of them at home and half in the office and no hotels.

Guggenheim has been a wonderful growth opportunity for me. I joined the core business when new companies were being acquired and have played several roles during the integration period.

For the first 9 years, I was the TFS administrator, during which I orchestrated all TFS upgrades, maintained user access, assisted users, organized TFS collections and projects, merged in source code from acquired companies and diverged source code to divested companies. I installed and configured Octopus as our standard deployment tool then mentored developers unfamiliar with Continuous Integration and Continuous Deployment (CICD) best practices so that everyone utilizes the same tools and process.

My current role is the Director of application development for the Treasury Services business. I have a collaborative relationship with the head of operations.  She shares her knowledge of the business and I share technical insight of the application and database.  Together, we have extensively refactored the database, cleaned up bad data, corrected reports, expanded data entry validation and improved performance.

 My objective from day one has been the elimination of production issues. I’ve achieved this by database and app code correctness, resiliency, simplicity and efficiency.

Beginning in 2018, I architected a new application utilizing WPF, C#, MVVM, and DDD to replace our three applications written in the early 2000s.  The application has a data entry focus, therefore WPF was chosen to avoid anomalies found in web and other non-native applications.  MVVM and DDD ensure the code is modular, readable, maintainable and extensible.  To make the transition to the new application seamless and at an early stage of development, the new application initially hosted the WinForms from the original applications alongside their WPF replacements.  A more efficient UI and highly desired features created a compelling reason for users to switch. My cross-application notifications feature became essential when everyone transitioned to working from home. When trades are added, removed or altered, a notification is sent to all users who have the application open.  These activities were commonly communicated verbally within the office.

To keep my team moving forward faster and avoid reoccurring mistakes, I have open discussions after our daily SCRUM meeting to share knowledge and discuss solutions to technical issues.  The first goal is to disseminate what each person has learned so that other team members may use it, making everyone a better engineer.  The second goal is to avoid teammates from getting stuck or implementing an inferior design by discussing the technical challenges and possible solutions.  We have created over 200 notes in OneNote to avoid losing this knowledge.  Further, I encourage teammates to do the research to find the correct solution.  My team has thanked me for my persistent drive for quality that has allowed us to eliminate all production issues and for challenging them to think outside the box and create something more than they thought possible.

I love learning, finding solutions, and being creative. I'm seeking to join a team where I can use these skills.  If you are interested in a self-motivated and highly competent engineer, let’s discuss how I can benefit your team.  Feel free to contact me.  Thank you for your time and have a great day.

 Sincerely,

      Jay Brockway

 JPBrockway@GMail.com

 847-778-9068

 https://JayBrockway.us