Hojin Nam

[E-mail](mailto:barojins@gmail.com) | [Github](https://github.com/barojin/) | [LinkedIn](https://www.linkedin.com/in/barojin/) | [Website](https://hojin.pro/)

**EXPERIENCE**

**GSCMS Inc, Seoul, South Korea Software Engineer Feb 2017 - Mar 2018**

* Create **task automation(dynamic-link library, C\#, .NET)** on Windows server with an open-source(selenium). The assembly reduced the time for the laborious task in the accounting dept, manually fetch bank statements of various online banking services, **from few hours to a second, and released to production.**
* Build the **test automation environment** in various Operating Systems (OS) on **VMware** for operational acceptance and regression testing in an exploratory approach. The corner, critical, and custom cases by analyzing error logs and defects from the beta users are considered. It **cut off the beta version error rate by 99%.**
* Integrate a financial management program via the interface with SAP, **create new UIs, modify legacy functions in the Windows app,** create the documentation for the main program according to various clients' needs.
* Build a **responsive web page** with AngularJS to support several **dashboards** in order to monitor server health and enable controlling the on-Premises service on the local server remotely for on-call team members.

**SKILL**

PROFICIENT: **Python, C#, React**, **Django, AWS, Git** | FAMILIAR: Java, JavaScript, HTML5, CSS, C/C++**,** SQL, NeDB,Node.js, Express.js, jQuery, MPI, CUDA, Spring MVC, Docker, VMware

**PROJECT**

* **O**
* [**Prouhet–Tarry–Escott problem on Message Passing Interface(MPI)**](https://github.com/barojin/PTEMPI/blob/main/cpi.c)**,** Implemented MPI parallel program parallelizing the code with 16 threads on a computer cluster and compared the performance of O(**2n**) computation between on single laptop and cluster computers**[C, Message Passing Interface library]**
* [**Matrix multiplication with shared memory on Compute Unified Device Architecture(CUDA)**](https://github.com/barojin/MMSMCUDA)**,** Optimized the use of memory to maximize the bandwidth on CUDA programming. It avoids redundant transfers from global memory. **[C++, CUDA Toolkit]**
* **Operating system:** [**Bounded-buffer problem**](https://github.com/barojin/PMC), Used the POSIX Threads(pthreads), semaphore to control access to shared resources. There are three functions such that producer which writes an element in buffer A, consumer which removes an element in buffer B, middleman which moves an element from A to B. **[C, semaphore, Ubuntu]**
* **Web System:** [**Web App with React.js, Node.js**](https://github.com/barojin/WRN)**,** Implemented CRUD table, role-based access control, session cookies on the front-end with Reactjs, encrypted password, REST architecture, test functions with Promise to manage asynchronous executions on the back-end with Nodejs. **[JavaScript, React, Node, NeDB, RESTful]**

**EDUCATION**

Cal State University East-bay, Hayward, CA **M.S. in Computer Science** Jan 2019 - May 2021

Sejong University, Seoul, South Korea B.S in Electronic Engineering Mar 2009 - Aug 2015

**COURSE**

Adv. Algorithms, Adv. Computer Networks, Adv. Theory of Computation, Adv. Software Engineering(Microservices), Web Systems, Cybersecurity, Parallel Computing(MPI, CUDA), Machine Learning, Capstone Examinations