# Geoffrey Meric

+33 6 51 26 20 96 | gcs.meric@gmail.com | linkedin.com/in/geoffrey-meric | github.com/gcsmeric | gcsmeric.github.io

## EDUCATION

**McGill University**                                                          Sep 2020 - May 2024
*Bachelor of Science in Computer Science, Minor in Finance*                          *Montreal, Canada*
- GPA 4.00/4.00
- Honors and Awards: Judie Rimer merit scholarship recipient, top 10% of cohort
- Relevant courses: Algorithms and Data Structures, Operating Systems, Software & Algorithm Design, Databases

## TECHNICAL SKILLS

**Languages**: Python, JavaScript/TypeScript, Java, C, HTML/CSS, SQL, Bash
**Frameworks and Libraries**: Node.js, Express.js, Jest.js, pandas, NumPy
**Technologies**: Git, shell scripting, REST APIs, GraphQL, Cloud Computing (AWS, GCP), Docker, Jira, CI/CD

## EXPERIENCE

**Software Development Intern**                                                 May 2023 - Aug 2023
*Autodesk*                                                                        *Montreal, Canada*
- Implemented real-time logging infrastructure in Python within an Autodesk Maya plug-in associated with a Node.js-based cloud rendering platform and engineered new workflows in the plug-in allowing for an approximately 3x increase in user rendering job submission throughput
- Introduced new UI elements for the plug-in using the Qt framework through PyQt, enhancing the user experience
- Developed a Command Line Interface (CLI) utilizing the Listr2 TypeScript library to enable simultaneous live monitoring of rendering job progress and automatic downloading of job outputs and logs as they're generated
- Designed a comprehensive Jest.js test suite to ensure robust validation of CLI outputs and downloads
- Deployed automated Node.js scripts to post-process auto-generated API client code files

**Software Engineering Intern**                                                 May 2022 - Aug 2022
*Procter & Gamble*                                                               *Geneva, Switzerland*
- Developed Python documentation tool to programmatically generate visualizations of the Pampers Customer Data Platform (CDP) architecture by interfacing with the Segment API via GraphQL, generating graphs using D3.js
- Deployed GitHub Actions workflows configured using YAML to sync CDP architecture changes in documentation
- Implemented monitoring framework for Braze Customer Relationship Management (CRM) platform operations by creating Python Airflow-automated data pipeline on Google Cloud Platform, interfacing with Braze REST APIs to fetch user change data and loading it to BigQuery, allowing for improved incident tracking and response capability
- Created web app using Node.js and Express.js used to automate CRM operations by leveraging Braze REST APIs, used as a proof of concept to design and send ad-hoc digital marketing campaigns in 20+ markets

**Software Development Intern**                                                 Jun 2021 - Aug 2021
*Teqoya*                                                                        *Paris, France (remote)*
- Collaboratively developed data visualization web application in JavaScript, PHP and HTML/CSS to retrieve, process and graphically display live air quality sensor data to test efficacy of air purification devices
- Utilized Chart.js library to display interactive data graphs and enable visual data comparison
- Implemented data analysis and downsampling capabilities in JavaScript, provided GUI streamlining improvements by implementing curve smoothing and data decimation capabilities

## PROJECTS

**Chrome Extension - RateMyProfs McGill** | *JavaScript, HTML/CSS*
- Developed JavaScript Chrome Extension that links professor *ratemyprofessors.com* pages in McGill university course websites
- Procedurally generated professor-specific queries using web-scraped instructor data
- Modified McGill webpage HTML Document Object Model to incorporate links with custom graphical interface
- Published extension to Chrome Webstore with over 30 users and 5/5 average rating

**Full-Stack Task Management Web Application** | *JavaScript, Node.js, Express.js, MongoDB, Google APIs, HTML/CSS*
- Built web application using Node.js and Express.js, stored and accessed application data using database deployed on AWS through MongoDB Atlas and performed CRUD operations through MongoDB API
- Interfaced with Google Calendar API using OAuth 2.0 protocol to enable users to import Google Calendar items