|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| EMAIL: | [jkotheimer9@gmail.com](mailto:jkotheimer9@gmail.com) | | | PORTFOLIO: | [https://jackkotheimer.com](https://jackkotheimer.com/) | | | | GITHUB: https://github.com/jkotheimer | | | | |
| Making life easier for other other developers since 2018, having mastered the art of simple & secure api creation and bash scripting automation. Searching for an opportunity to create a neural network | | | | | | | | | | | |
| Languages | | | Frameworks | Databases | | | Tools | | | | |
| * Java * C * C++ * PHP | | * Python * SQL * JavaScript * Bash | * Apache CFX * Node.js * Flask * Django | * MySQL * Amazon RDS * MongoDB * Firebase | | | * Git * Linux * AWS/ ECS/EB * Docker compose | | | | * SSH * Nginx + Apache * Celery/RabbitMQ * OpenAPI Spec |
| Education | | | | | | | | | | | |
| Loyola University Chicago | | | B.S. Software Engineering (Deans List) | | | May 2020 | | | | | |
| Work Experience | | | | | | | | | | | |
| Full-Stack Developer | | | EzClinic | | Chicago, IL | | January 2021 - Current | | | | |
| * Wrote documentation outlining the REST API specification for new server logic that would maintain prior functionality * Completely decoupled server logic from client apps by refactoring the codebase to a REST API written with Django Python   + Implemented the decoupling of each HTTP request one CRUD function at a time, for both the front and back ends of the app * Utilized previously written script to deploy each update on a staging server to be tested by QA before being pushed to production * Currently:   + Researching multidimensional neural networks to begin work on a medical diagnostic machine learning algorithm | | | | | | | | | | | |
| Back-end Developer Intern | | | EzClinic | | Chicago, IL | | September 2020 – January 2021 | | | | |
| * Reorganized source code repository and filtered out unnecessary and unsafe files and wrote bash scripts to automate the 3 main deployment mechanisms:   + Local development & debugging server deployment   + Remote staging server deployment using ElasticBeanstalk & test result fetching   + Production server deployment using ElasticBeanstalk & CNAME swap * The script builds a Docker image for each of the above servers from a templated Dockerfile, which is pushed to Dockerhub for ease of use and version control   + Built a celery notification system running on a RabbitMQ server into each Docker image   + Configured MySQL, RabbitMQ, and Nginx servers and on each Docker image   + Wrote several deployment hooks to run the Django wsgi on Nginx, connect it to the MySQL server, & connect a Celery notifier to the RabbitMQ server * Executed the migration from a local database to a remote database with a secure authentication of credentials upon deployment | | | | | | | | | | | |
| Computer Science Tutor | | | Loyola University Chicago | | Chicago, IL | | November 2019 – May 2020 | | | | |
| * Tutored Loyola Computer Science students on an appointment basis * Ensured high quality aid to all students both during and outside work hours | | | | | | | | | | | |
| Research Assistant | | | Loyola University Chicago | | Chicago, IL | | July 2019 – December 2019 | | | | |
| * Analyzed videos of CPS courses, ensuring curriculum and research standards were upheld * Participated in a series of seminars determining the most effective educational techniques for computer science | | | | | | | | | | | |
| Java Programming Instructor | | | Digital Media Academy | | Chicago, IL | | June 2019 – August 2019 | | | | |
| * Taught fundamentals of the Java programming language to high school students * Structured a curriculum with hands-on project based learning | | | | | | | | | | | |
| Android Research Developer | | | Loyola University Chicago | | Chicago, IL | | May 2019 – June 2019 | | | | |
| * Collaborated with three other software engineers in person, full-time for a month * Created customizable sign-language-based app for physically and vocally disabled people (see ‘TalkMotion’ under Project Experience tab) | | | | | | | | | | | |
| Projects | | | | | | | | | | | |
| Pianoboard | | | [https://jackkotheimer.com/projects/#pianoboard](https://jackkotheimer.com/projects/" \l "pianoboard) | | | | | | | 2018 - Current | |
| * Wrote a RESTful web service to create music projects which store and process client-made MIDI recordings * All CRUD operations occur in an isolated Node.js API, completely decoupled from the web page server * Web pages rendered with PHP running on an Nginx server. UI CRUD implemented in JavaScript, using XHR requests to the API * Created an opensource deployment mechanism for Docker based applications, using docker-compose and ecs-cli for local and remote deployment respectively * **Utilized**: PHP, Node.js, Nginx, HTML/CSS/JavaScript, MySQL, Docker, Docker Compose, AWS ECS, Bash, Git | | | | | | | | | | | |
| ChatApp | | | [https://jackkotheimer.com/projects/#chatapp](https://jackkotheimer.com/projects/" \l "chatapp) | | | | | February 2020 – May 2020 | | | |
| * Developed a customer service based chatting application with 3 other developers * Team accomplishments:   + Pieced together 4 developers’ code, mixed betweeen back-end and front-end, into a functioning chat app   + Communicated entirely through Slack messages to orchestrate the app’s construction * Personal contributions:   + Wrote Python code using the Flask framework to create the user dashboard, chat request/approval, and chat room C.R.U.D.   + Created Jinja2 templates to render user and chat info for the dashboard and chatroom web pages   + Automated the building/deployment of the development server with Bash scripting   + Deployed the finished product to an AWS EC2 instance for final presentation * **Utilized**: Python, Flask, Jinja2, HTML/CSS/JavaScript, MySQL, Bash, AWS EC2, Git | | | | | | | | | | | |
| E-Commerce API | | | [https://jackkotheimer.com/projects/#ecommerce](https://jackkotheimer.com/projects/" \l "ecommerce) | | | | | | August 2019 – December 2019 | | |
| * Built a REST API for a barebones e-commerce website with 2 other developers * Team accomplishments:   + Used GitHub as a workflow efficiency tool to merge and test frequent code updates from peers * Personal contributions:   + Wrote all CRUD operations for products, and order execution   + Created a simple UI with HTML/CSS/JavaScript to provide interface for the API * **Utilized**: Java, Apache Server w/ CXF, Apache Maven, Junit, MySQL, Hibernate, AWS, Git | | | | | | | | | | | |
| TalkMotion | | | [https://jackkotheimer.com/projects/#talkmotion](https://jackkotheimer.com/projects/" \l "talkmotion) | | | | | May 2019 – June 2019 | | | |
| * Created an Android app to translate sensor readings from hand gestures to text-to-speech commands with 3 other developers * Team accomplishments:   + Worked at a full time capacity in close quarters for a month to produce a faux machine learning system   + Distributed work evenly among each developer with different backgrounds and experiences * Personal contributions:   + Created the SQLite model for “training” the computer to recognize gestures   + Co-wrote the algorithm to recognize a motion based on 2+ training gestures (ask about it!) * **Utilized:** Java, SQLite, Android Studio, object state machines, machine learning, Git | | | | | | | | | | | |