

# HEKMAT HAMIDI

SOFTWARE DEVELOPER - TORONTO, ONTARIO

✉ hamidihekmat@gmail.com  
in <https://linkedin.com/in/hamidihekmat>  
☎ (416) 428 - 2923  
🌐 <https://hekmathamidi.me>

## SKILLS

### Languages

- **JS | Typescript** created various web app projects PERN Stack
- **Python** and Selenium used to create multi purpose web scraper
- **C | C++** used to develop assembly line system OOP for college finals project

### Tools & Frameworks

- **ReactJS | NEXTJS** to build reusable UI components (Styled CSS Components) and fully functioning web apps
- **NodeJS** with ExpressJS used to deploy backend secure API services (REST & GraphQL)
- **Docker** for managing, containerizing, and deploying application services
- **LINUX | UNIX** for managing services, security firewalls, SSL, and DNS records
- **GIT** for source code version control and coordinating tasks amongst colleagues
- **Figma** used for prototyping designs, wire-framing, and creating social media events

### Databases & IaaS

- MySQL, PostgreSQL, SQLite, MongoDB, GCP & AWS

## EDUCATION

### Seneca College

Advanced Diploma In Computer Programming & Analysis  
Sep 2019 - 2022

### Seneca College

Associate's Degree In Systems Networking  
2017 - 2019

## EXPERIENCE

### WRCCDC | Software Developer Remote | California, US

September 2021 - Present

- Rewrote and containerized **Operations Bot** in Python using SQLite, Docker and Discord APIs, reducing administration tasks by **30%** for Western Regional Collegiate Cyber Defense Competition
- Automated provisioning of scoring engine data using job scheduler to trigger real-time event notifications via **WebHooks**
- Leveraged **JSON Web Tokens** to streamline dynamic channel routing and permission management for authenticated users

### End Group Development | Automation Programmer | Toronto, CA

April 2019 - May 2020

- Programmed and installed smart home IoT devices, automated lighting and security systems for residential housing
- Implemented scalable network infrastructure and **Network Failover** strategies for businesses
- Improved existing sales proposal for potential customers by providing detailed work procedures and planning processes resulting in **75%** customer retention

## PROJECTS

### Void Encode

- Designed and developed robust HLS video encoding architecture using **BullMQ** simple messaging queue, NodeJS, Ffmpeg, Redis and Typescript to distribute encoding tasks across multiple worker nodes
- Implemented an event based notification system to track progress of transcoding jobs
- Used **AES** algorithm to encrypt HLS segments restricting public access to transport stream segments... [demo](#)

### NOS

- Developed a video streaming web app **PERN** that can handle millions of concurrent viewers using Google Cloud Storage
- Integrated multiple external APIs such as TMDB, TVDB and Trakt to create dynamic UI components with **NextJS**
- Leveraged Cloudflare's **Edge Network** to cache static assets and frequent JSON response using **stale-while-revalidate** caching strategy resulting in **75%** overall site performance... [demo](#)

## CERTIFICATION

### Control4 Technician | Automation Programmer | Chicago, US

May 2019 - No Expiration Date

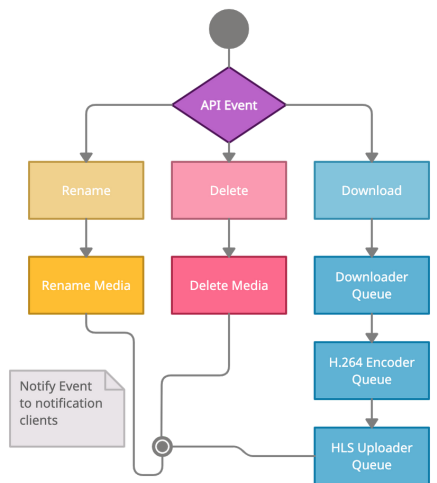
- Attended 7 day hands-on practical automation programming training and obtained **Control4 Certified Dealer** title
- Achieved a **96%** test score for the final assessment

# HEKMAT HAMIDI

SOFTWARE DEVELOPER - TORONTO, ONTARIO

✉ hamidihekm@gmail.com  
in https://linkedin.com/in/hamidihekm  
☎ (416) 428 - 2923  
🌐 https://hekmathamidi.me

## VOID ENCODE (SCALABLE ENCODING SERVICE)



### What?

- Designed & developed a scalable video encoding architecture
- Implemented simple messaging queue **BULLMQ** to distribute jobs across multiple worker nodes

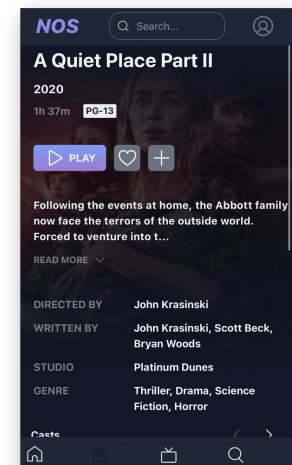
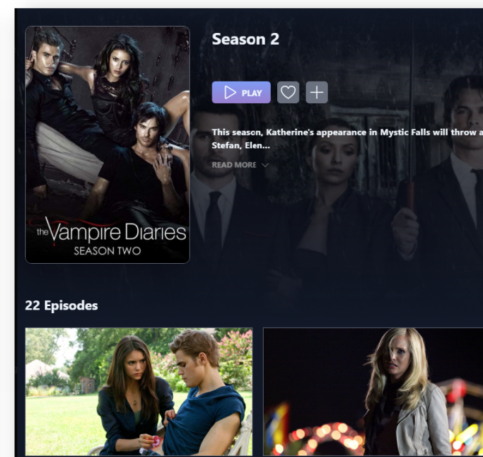
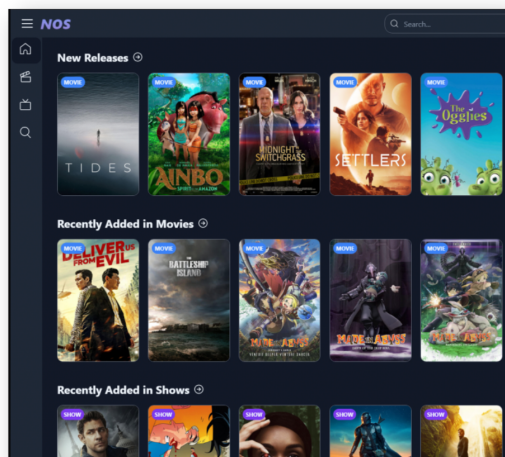
### How?

- FFMEG to encode video format to H.264
- Dashboard to track worker's job progress
- Cached file segments for efficient and quick access

### Results

- HLS video stream playable across **99%** web browsers & devices
- Encrypted TS segments for access to only authenticated users
- Private code, request for access... Explanation of the project [here](#)

## NOS (STREAMING WEB APP)



### What?

- Responsive streaming web app **PERN**
- Reusable styled UI components with **ChakraUI**
- Video playback Trailer(s) with **React Player**

### How?

- Designed with **Figma**, built with **React**
- TMDB **APIS** to create dynamic & **responsive** UI components
- Implemented infinite scroll **pagination** using react **custom hooks**
- Cached** API responses with **SWR**

### Results

- Beautiful **usable** web app for Movies& TV Shows
- Performant/Efficient due to **Edge & Browser caching**
- Deployed on a live domain, visit [here](#)