Dhilip Kumar

Software Developer

Profile

Self-directed and motivated Problem solver with Fluency in Python, NodeJS, Angular. Experience in building web application, micro services, system design and architecting with AWS cloud with additional knowledge of Data Engineering, Machine learning techniques.

Education

Graduate Diploma In System Analysis, Institute of Systems Science, National University of Singapore, Singapore

January 2019 — February 2020

- Graduated with 4.13 GPA

Bachelor of Engineering: Mechanical Engineering, University College of Engineering, BIT Campus , Tiruchirappalli

January 2012 — January 2016

- Graduated with 8.13 GPA
- Received Gold medal for academic excellence of holding 1st rank in Department of Mechanical and Automobile studies
- Dean's Honours List Semester 8 Year-2016

Employment History

Software Developer at Echol Tech Pte Ltd, Singapore

May 2020 — Present

1. **Changi Airport Group – Vehicle Detection and Queue Formation**
    - Dwell time computation for each truck passing through the gantry.
    - Built a seamless pipeline using Python, to process all incoming frames from the video stream in real-time **[RTSP]** to provide timely congestion metrics.
    - **Flask** Python Micro Service: Object detection using **Yolov4 Neural Network -** Perform object detection once (or once every *N* frame) to identify trucks. Tracking of the object using **dlib** (correlation tracking algorithm) as it moves in subsequent frames without having to perform object detection.
    - Built a Real Time Dashboard for Traffic Congestion Alerts using **Angular, Nodejs** and **MongoDB**.
    - Validating identified objects against the targeted zone (area of interest)

Details

# 27 - 527 116A Jalan Tenteram,
321116, Singapore.

+65 90890177

Links

dhilip_kumar@u.nus.edu

linkedin.com/in/dhilipbinny

github.com/DhilipBinny

Skills

Python

Git

Docker Swam, Docker Compose

MySQL, MongoDB, SQLite, Redis

Node.js

Docker API

Amazon AWS

Azure CI

CLI -Bash, Bash Script

EC2, Lambda, ALB, SNS, S3

Data Visualization : Matplotlib, Seaborn, Plotly

Machine Learning Techniques

Kubernetes

IDE : Visual Studio Code, PyCharm Jupyter Notebook, Colab

Framework : Flask, ASP .net MVC Dialogflow

Web Development : Angular,

HTML, CSS, JavaScript

2. **Housing & Development Board - East Staging Ground Monitoring System**

- **Weighbridge Agent –** Developed and deployed NodeJS API which integrates with hardware such as PLC and Weigh Scale (through serial ports), Printer and IP cameras (RTSP streams).
- Configured **MySQL master master replication** between physical enterprise servers at different location, in order to provide redundancy and also to prevent data loss.
- Implemented a **Python** microservice to monitor DB replication status. Auto resolve known/identified replication issues (*Duplicate Entry for Primary key* and *Key not found* errors). And auto send alerts to **slack channel** regarding the replication status including replication lag , errors and disconnection between master and slave servers.
- Implemented a Python Cron Scheduler to schedule an automated **MySQL dump backup** sync into SFTP Serve along with custom retention policy - preserve n days.
- Developed **Grafana** dashboards to monitor app server nodes and MySQL nodes to display the system metrics collected by **Prometheus**.
- Docker swarm deployment
    - Containerizing the web application service – Creating and configuring **Dockerfile**.
        - Frontend Container - NGINX container to server the Angular client app with custom configs [loading ssl certs, reverse proxy configs]
        - Backend Container - .NET core container which serves as the API for the web app.
    - Created the docker compose yaml file to configure the application's services.
        - Environment variable injection to the containers
        - Mounting external volumes and networks
        - Defining restart policy for the services.
- Coordinated feature requests with Client.
- Supervised work of fellow developers, assigned tasks and monitored performance against targets.

3. **LTA – Face Recognition & Biometric Attendance system.**

- **Face Recognition Engine - Flask**
    - Developed the face recognition engine using **TensorFlow** and **face-recognition** with following endpoints
        - Training, Recognition and Image Quality Check (mask, brightness, contrast, face blur).
    - Auto load user's face-encodings stored in S3 bucket on container start/restart.
- **Face Engine Router – Flask**
    - Python microservice to check the health of containers running in multi AZ EC2 hosts
    - Single API Endpoint to update the face-embeddings to all the containers while training, retraining or deactivating a face
- **AWS cloud architecture set up and deployment**
    - Migrated the legacy attendance reporting application from on-premise VM to AWS Cloud
    - Managed to Setup a **Azure CI pipeline** to build docker images, test it with **sonar cloud** code coverage and deploy as docker containers in AWS EC2 nodes.
    - Architecture and cost estimation of the relevant AWS resources.
    - Setup **Jump Host** nodes, custom **WAF** rules to control/restrict requests based on geo locations as per security requirements.
    - Configured Application Load balancers to route traffic across **Multi AZ** target groups.
    - Implemented **Lambda** functions triggered by CloudWatch scheduled events.
    - Setup CloudWatch triggers and alarm based on CloudWatch logs.

    - Setup **SNS** alerts for notifications via email for **CloudWatch** triggers and **Lambda** function status.
    - Setup agents in **EC2** machines to collect **custom metrics**.

4. **Natflow Pte Ltd - IOT Dashboard**
    - Developed a **MEAN – Angular, Nodejs, Mongo dB** stack application which monitors the Airflow measurements across various Air Handling units and Hot water recovery system
    - Implemented a python script for collecting sensor data across multiple Modbus channels and pump it to **MongoDB.**
5. **National Environment Agency (NEA)**
    - Implemented a **NodeJS** based microservice that restreams the **RTSP** feeds from multiple IP cameras and serve them as **WebSocket** to the client.
    - Implemented a scheduler microservice **python script (multi-threaded)** to take snapshot images from multiple IP cameras at regular intervals.
    - Implemented a proxy request-forwarder microservice which
        - Listen to incoming requests.
        - Inject headers with digest authentication and send the requests to the IP cameras (to perform PTZ actions).
        - Transmit the response as it is.

Internships

Software Developer at A*STAR - SIMTECH, Singapore

September 2019 — January 2020

- Designed and Developed Name Card Scanner for Auto Populating Web-form from Images using **Python**, **OpenCV**, **Pytesseract(OCR),** and **spaCy(NLP)**.
    - Built a 3-Tier web application with **Angular** frontend and Flask API Server backed by **MongoDB**.
    - Applied regex to extract website, email, contact numbers. Used Named Entity Recognition module from Spacy NLP to extract names from raw text.
- Built a Real Time Dashboard using Angular and Nodejs for Capturing the Order status processed by a Robotic Arm.
    - Demonstrated the Dashboard at ITAP (Industrial Transformation ASIA-PACIFIC) a leading trade event for Industry 4.0 !
- Designed and Developed an Event Management System using **Angular**, **Nodejs** and **MongoDB**.
    - A whole functional web Application for Event Creation, Event Registration, Notification, Check-in and Bar-code Printing.
    - Took Part in Design and Implementation of complete SDLC, Integration and Testing.

Project Experience

1. Real-time data analysis project on CDN response time

    · Setup a Kinesis data steam on AWS backed by S3 bucket.

    · Used Athena to query data from S3 and perform on demand data analysis.

    · Generated reports on CDN performances across regions and different time intervals.

2. Park Vehicle Movements volume prediction and forecasting

· Transformed the flat file into graph dataset using Networkx and conducted EDA to understand the traffic pattern across the park.

· Build a forecasting model using prophet library to predict the vehicle volume across gates.

Courses

AWS Certified Cloud Practitioner - https://bit.ly/3gqUHIB

February 2021 — February 2021

Data Analyst with Python DATACAMP - http://bit.ly/2qofB3x

September 2019 — September 2021

Data Scientist with Python DATACAMP - http://bit.ly/35hieDn

September 2019 — September 2021

Python Programmer DATACAMP - http://bit.ly/2QuSCOT

August 2019 — August 2019