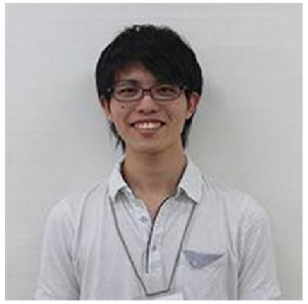
若手エンジニアが育成に携わってみた@AITAC

自己紹介

西村優吾



- ➔ NTT コミュニケーションズ 技術開発部
- ➔ 主な業務：ネットワーク自動化
- ➔ 沖縄オープンラボラトリ主催スペシャリスト育成プログラム第1期生

Agenda

- AITACとは
- Step1のカリキュラム紹介
- NTTコムのAITACへの期待/関わり
- 私が担当した講義
- AITACで講師をするということ/得られたこと
- フルスタックエンジニアを育成するには
- おわりに

AITACとは

インフラ人材育成

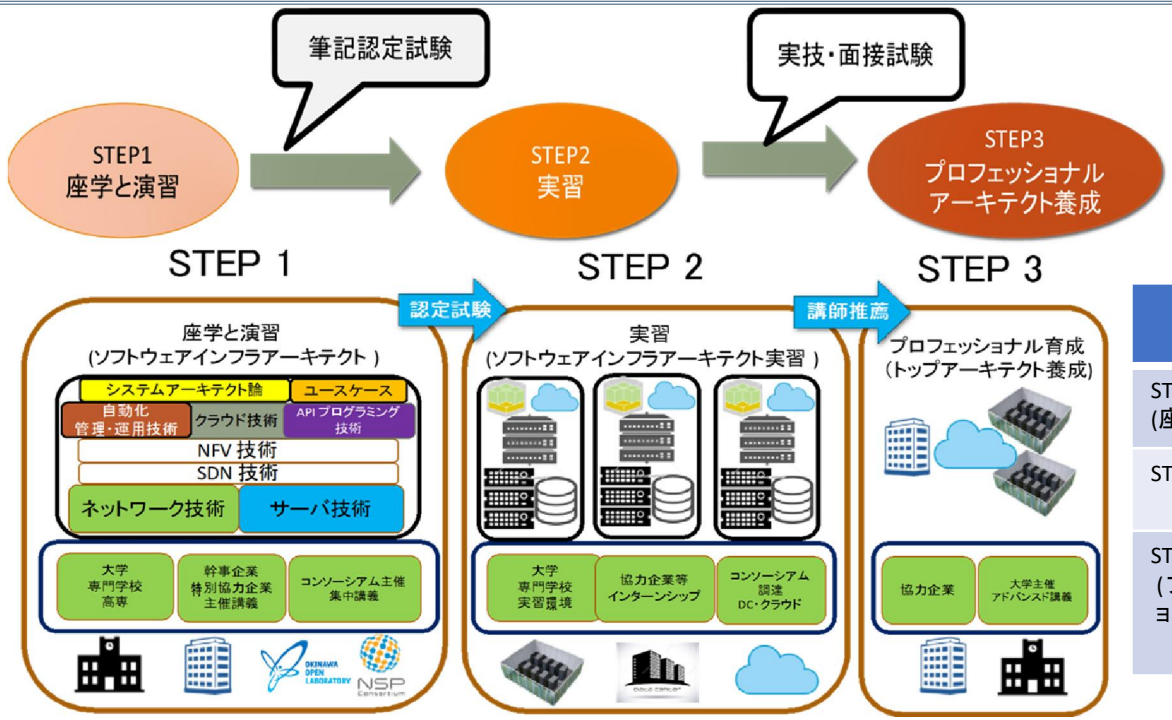
- 高度 IT アーキテクト育成協議会
  AITAC : Advanced IT Architect Human Resource Development Council

https://AITAC.jp/
contact@aitac.jp

AITAC が育成を目指す人材像

- トップアーキテクト
  - ICT 技術を「武器」としてアイディアをシステムに具現化できる人材
- フルスタックエンジニア
  - 従来の区分である「ネットワーク管理者」と「サーバ管理者」の垣根を超えたハイブリッドなエンジニア
- システムを運用できるエンジニア
  - システムを理解しトラブルを解決できる人物
  - 監視を行うことのみが運用ではない

人材育成のステップ



- 段階をふんだ学習体制
- 段階毎の認定制度
- 実践的シナリオに基づいた
  グループ学習制度

履修段階	学生のケース	社会人のケース
STEP 1 (座学・演習)	半期(15週) 授業 x 2	集中講義(3日) x 4
STEP 2 (実習)	集中実習(5日) x 3	集中実習(5日) x 3
STEP 3 (プロフェッショナル育成)	インターンシップ (1ヶ月) Interop Tokyo への参加	人脈形成をサポート

Step1のカリキュラム紹介

Step1のカリキュラム

- Network基礎からSDN,OpenFlowまで
- OSとLinuxの話(Server周り)
- NTPやDNS
- ストレージとDBの話
- クラウドという概念から利用法
- Network・Server運用監視
- 自動化
- コンテナ
- セキュリティ

STEP1 (ネットワーク)

ネットワークの基礎技術 構築・運用技術 設計方法	講義の概要 ネットワークの理論	本講義の位置付け、実施方針を伝える
	TCP/IP ネットワークの概要 Ethernet の仕組み	TCP/IPネットワークの全体像と階層モデルについて理解する
	Ethernetを用いたネットワークの構築演習	Ethernetを用いて小規模なネットワークを構築する スイッチの操作を覚える
	ネットワーク層 IP経路制御 (1)	ネットワーク層の役割について理解し、IPアドレスの構造、IGPの仕組みを理解する
	IP経路制御(2)	EGPの役割とBGPの仕組みについて理解する またBGPを応用した実際の経路制御手法についても学ぶ
	経路制御演習 (1)	IGPとEGPの階層的な経路制御について学び、RIP、OSPFを用いた経路制御の演習を行う
	経路制御演習(2)	BGPの仕組みについて理解し、ルータの設定を通じて理解を深める
	ネットワーク冗長化と設計	信頼性向上のためにネットワークの冗長化が必要であることを理解する 各層での冗長化技術を学ぶ
	ネットワーク設計論	データセンタや ISP における実際のネットワーク設計や Interop Tokyo におけるネットワーク設計からその意味を学ぶ

STEP1 (コンピューティング)

サーバの基礎技術 **スケーラビリティー** **仮想化技術**	OS/コンピュータアーキテクチャ	サーバに用いられるOSの基礎的な仕組みを理解する 現在のコンピュータアーキテクチャとCPUの仕組みについて学ぶ
	OS/サーバの仕組み セキュリティ	Linux を例として OS の構造と仕組みを理解する Linux を利用して構築されるサーバの事例を理解する
	Linux のインストールと サーバとしての設定	OSインストールからソフトウェアのインストール、性能試験までを一通り行い、サーバ構築の一連の流れを学ぶ
	負荷分散と仮想化	サーバ仮想化技術について学びその利点と欠点を理解する サービスの負荷分散手法について学ぶ 実際のシステムについて事例を通じて理解を深める

STEP1 (クラウド・仮想化技術)

最新ネットワーク制御技術 サービス仮想化 統合管理技術	大規模システム構築に向けて	サーバ仮想化や負荷分散、ネットワークの冗長化を利用して大規模可能なシステムの構築について学ぶ
	クラウド技術の概要 クラウドサービス構成法	クラウドのアーキテクチャとその要素技術を理解する
	ストレージとデータベース技術	ファイルシステムからネットワークベースのストレージ、データベースの技術について理解する。
	コンテナ技術と演習	オンプレミスクラウドの構築について、IaaSとPaaSの場合を経験する オンプレミスクラウドを構築する場合の注意点とセキュリティについて学ぶ
	商用クラウドを用いたサービスの構築	商用クラウドの利用方法を学び、商用クラウドを利用したシステム構築事例について学ぶ
	商用クラウドサービスの活用 ハイブリッドなシステムの構築	商用クラウドを利用して Web サービスを構築する手法について学ぶ データベースを利用しデータ解析基盤として利用する手法について学ぶ
	SDN技術	SDN技術の基礎を理解し、その特徴と既存ネットワークとの違いを理解する
	OpenFlowを用いたネットワークアプリケーションの作成	OpenFlowアプリケーションを実装できるようになる
	NFV技術の概要	NFV技術の概念を理解し、その実現方法と技術課題について理解する

STEP1(サービス設計・自動化技術)

各技術を適切に活用し サービスインフラの 設計・構築・プログラミング を用いた設計理論を習得する	インフラ設計論 構築プロセスの紹介	サービスを構築するにあたって必要な要件とそれを満たすコンポーネントをどう利用するか、議論を通じて理解を深める ShowNet や大規模ネットワークを事例としたネットワーク設計論を紹介する
	インフラ運用・監視・管理技術	構築したインフラシステムを監視するための手法と、監視要件を元にした監視システムの構築について事例を通じて学ぶ
	サービスインフラのセキュリティ	サービスインフラに対して求められるセキュリティとその注意事項について実際の事例や教材を元に学習する
	インフラ構築と運用の自動化 (必要なプログラミング初歩を含む)	インフラ構築を自動化するためのツール群に関して、プログラミング言語の初歩を交えながら学習する
	運用ツールによるインフラ構築と運用の自動化	自動化ツールを用いてインフラの構築と運用自動化を体験する
	実シナリオに基づいたアーキテクチャ設計	グループを形成し、グループ単位で与えられたサービス課題を実現するためのシステムアーキテクチャについて議論を行いまとめる

NTTコムの
AITACへの期待/関わり

NTTコムとして取り組むモチベーション

- 人材育成
  - ソフトウェアからインフラまで体系的に学べる数少ない本カリキュラムにおける、講座受講や講師派遣による社員のスキルアップ
  - ユーザ企業エンジニアや営業まで幅広く育成することによる、提案の円滑化や利用範囲の拡大・深化
- 人脈形成
  - AITAC会員企業や教育機関との交流

NTTコムにおける技術育成

- 入社時に、少しだけ技術研修がある(今年は5日)
  - 今年は著名な方々を外部講師として招いた
- 技術開発部ではbootcampという育成制度がある
  - NW/Sec/IoT/Cloud/App/DSAI　の分野でそれぞれ若手が一年目に研修する

コンテナ技術と演習
(私が担当した講義)

私が担当した講義

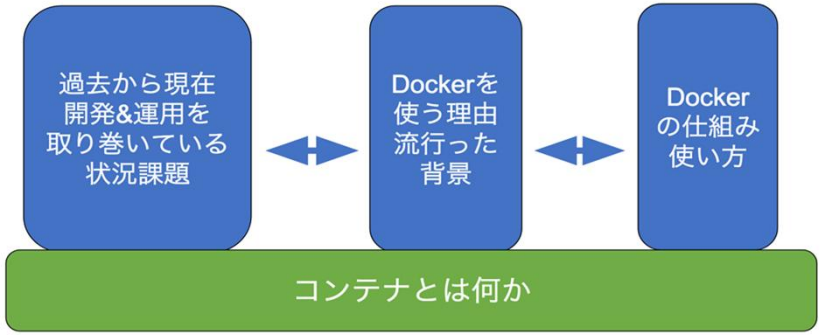
- 「コンテナ技術と演習」という題で
  講義と演習を行った

今日のGoal

- コンテナ技術の基礎が分かるようになる
- **コンテナ技術の隆盛の理由を理解し、正しい場面でコンテナを選択できるようになる**
- Dockerに触れてみる(ハンズオン&演習)

高度ITアーキテクト育成協議会 3

本講義の軸

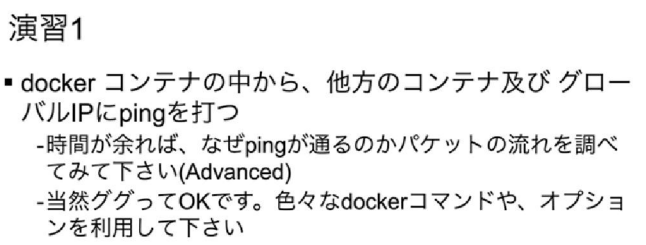


高度ITアーキテクト育成協議会 4

講義の様子

講義を作る際に注意したこと

- 多くの受講者に満足してもらいたいというモチベーション
- 受講者のバックグラウンドが非常に様々という問題
  - 講義：コンテナの基礎からname spaceの説明まで広い範囲で講義を行い、いろいろなレベルの人に対応
  - 演習：ハンズオン(コピペで動く)と演習(考えないと解けない)に分けることで、幅広い層に対応



AITACの講師経験を通して

AITACの講師経験を通して

- 講師として、受講者のレベル感に合わせた内容や座学・ハンズオンの構成を考えることを意識できるようになった
- カリキュラム内容の策定から講師経験を通じて、育成側の難しさと責任感を強く意識
  - 学生への技術広報活動を通して、ソフトウェアもインフラも両方興味あるという人は少ないことを再認識
  - そもそもこのカリキュラムに参加してくるだけで両方に興味を持っている人が集まっている

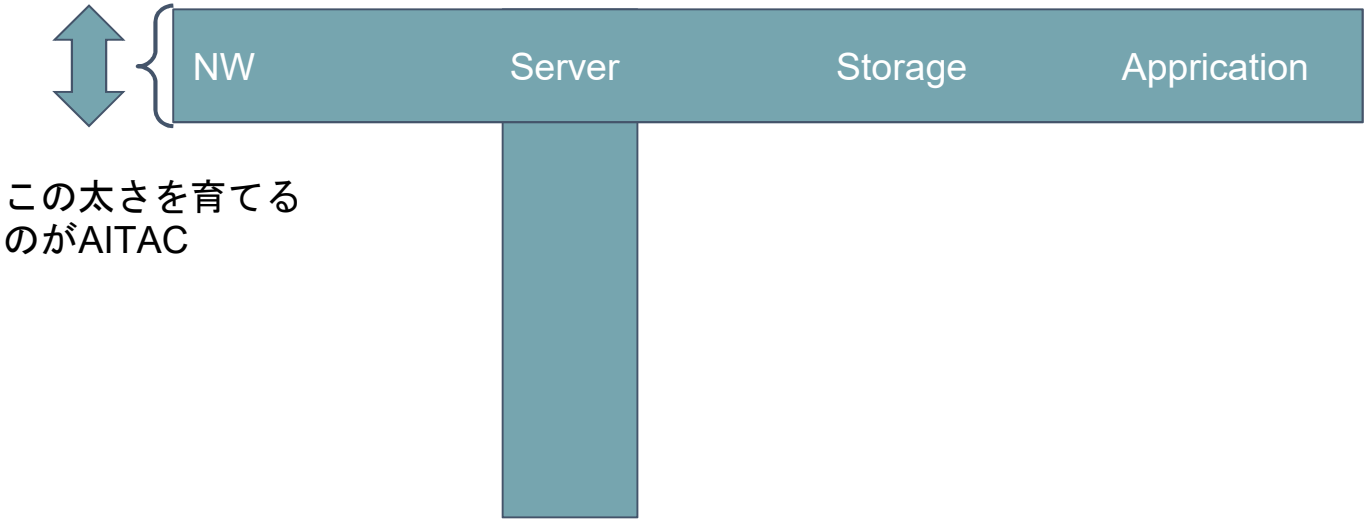
フルスタックエンジニアを育成するには

フルスタックエンジニア

- Network/Server/Storage/Applicationに精通するエンジニア
  - そんな人そうそういない
  - すぐ育てられるわけもない
  - AITACに参加していただける企業の中からどうやってこういう人を育成していくかを常に考えないといけない

フルスタックエンジニア

- よく言われるのはT字型/π型人材



おわりに

AITACに関わってきた率直な感想

- 受講者目線
  - フルスタックエンジニアになるための幅広い学習の第一歩としてカリキュラムの構成が優れていると思った
  - 各セクションのエキスパートがweb系でもEnterprise系でも必要とされるスキルを厳選しているのがよい
    - 同時に、講師の技術への愛が顔をだしており、人によって講義の難易度にばらつきがあるという課題は抱えている

最後に

- AITACのカリキュラムを通してフルスタックなスキルを学んでいくきっかけとなって欲しい
- 当然教える側としても、担当範囲だけでなく幅広く勉強していこうというモチベーションにもつながる
- カリキュラムを通して受講者や講師でのコミュニティができ、それぞれがエンジニアとして成長していきたい

STEP2の概要

STEP2の概要

約1ヶ月間、業界で活躍するインフラエンジニアや学術機関の研究者などの最先端の動向に詳しいメンターから指導を受けながら、AITACにて設定した課題にグループワークで取り組んでいただきます。

●課題の取り組み方
- 応募者同士で2～3名程度のグループを形成して行うグループワークとなります
- グループワークは主にオンラインチャットを用いて行われ、1ヶ月の期間において6日～8日程度の時間を目安に、時間を割いて参加していただきます。

●課題を取り組む期間
1ヶ月程度

STEP2の概要と大まかな流れ

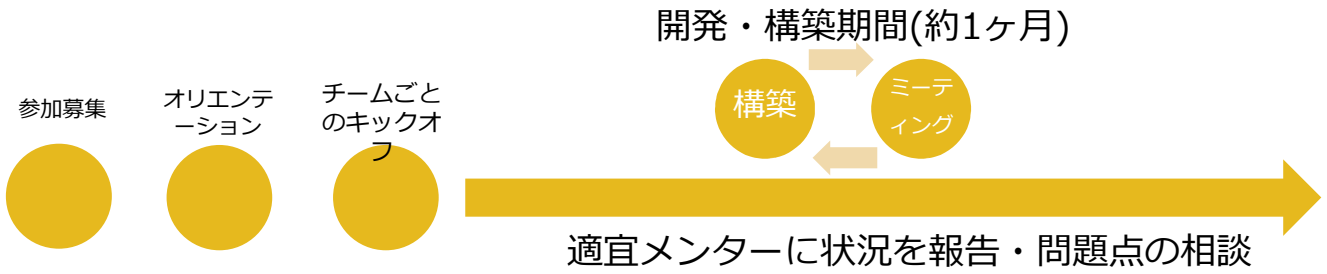
●取り組む課題について
課題は大きく分けて「初級」「中級」「上級」の3つの難易度が設定されております。
「初級」の課題から取り組んでいただき、「中級」「上級」と順番に難易度に取り組んでいただきます。

●STEP2全体の流れ

初級から始めていただき、上級課題をクリアすることでSTEP2認定となります

STEP2課題に取り組む大まかな流れ

●課題一つ実施するにあたっての全体の流れ



各チームでのキックオフ後、各チームごとに課題の計画・設計・構築とミーティングなどを繰り返しながらシステムを構築、最後に構築したシステムをプレゼンしてもらい、全体の評価を行って修了となります。

初級課題

社内コミュニケーション サイトの構築

課題7

- タイトル： 社内コミュニケーションサイトの構築
- 課題の内容： オープンソースのソフトウェアを活用して社内コミュニケーションサイトを構築してください
- 課題の要件：
  - 機能をサーバ（VMを使用）にインストールして社内LANから利用できるようにする。
    - チャット機能
    - ファイル共有
    - 社内連絡用のWebサーバ
  - VM一つの中に構築すること
  - 使用するソフトウェアはOSSを使用すること
  - チャットツール・ファイル共有についてはユーザ管理を行えること
  - Webサーバについては複数の部署からの要求で違う環境をそれぞれ構築することを想定すること

前提条件

1. 企業規模(ユーザ数)
   - 中規模企業(資本金3億円以下 300人以下)
   - 中規模企業の平均は40.1人 = 従業員数は**40人**程度と想定する。
2. 機密データの管理方針の想定
   - オンプレミス、または自身でデータの管理ができるAWSには機密情報を置いても良いこととする。
   - slackはslack社にデータが管理されてしまうため利用NGとする。

2017 年版 中小企業白書 (中小企業庁)

目標の設定

- 課題全体
  - 必要な要件を満たしつつ、新しい技術を学ぶ
- チャット機能
  - メールに代わるコミュニケーション環境を作る
  - Slackが使用できないのでSlackに代わるサービスを提供する
- ファイル共有
  - 自分たちでデータを管理しつつ簡単・安全に情報を共有する
- 社内連絡用のWebサーバ
  - 様々な部署が手軽にWebサーバやWebアプリケーションを利用できる環境を作る

要件定義(サービス基盤)

基本はすべてAWS上に構築する

- 機密データ管理条件をクリアできる
- イニシャルコストが低い
  - 物理スペース・電力・冷却・サーバなど
- 物理作業が必要ない
- スケールアウトがしやすい
- 便利なAPIやサービスを利用できる

使用したサービス

要件定義(提供するサービス・機能)

1. チャット機能
   a. ブラウザベースのビジネスチャットツールをサービス提供する
   b. ユーザ認証機能を提供する
   c. 可用性を確保する
2. ファイル共有機能
   a. ブラウザベースのファイル共有ツールをサービス提供する
   b. マルウェア検知・解析機能を搭載する
   c. ユーザ認証機能を提供する
   d. 可用性を確保する
3. 社内連絡用のWebサーバ
   a. 複数のCMS（ブログ,wiki,SNS）をサービス提供する
   b. SOシステムを提供する
   c. XSSなどの攻撃緩和(WAF)の導入
4. VPNサーバ
   a. AWS環境と社内・リモート環境を安全に結ぶ

提供するサービス

Chat

File Share

Wiki

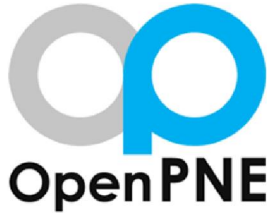
Blog

未提供

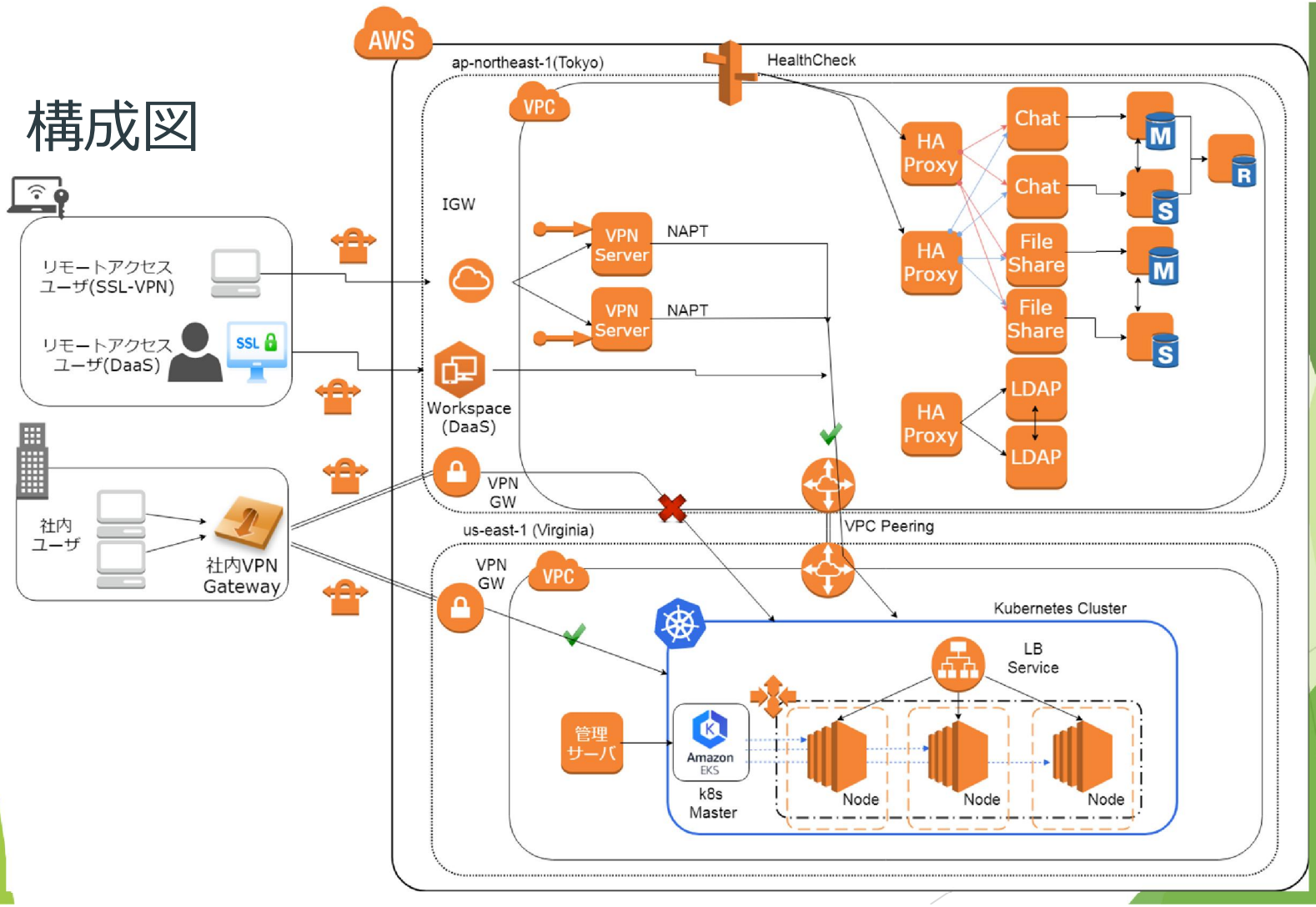
VPN

SoftEther VPN

SNS



構成図


AWS
ap-northeast-1(Tokyo)
HealthCheck
VPC
IGW
VPN Server
NAPT
VPN Server
NAPT
HA Proxy
HA Proxy
HA Proxy
Chat
Chat
File Share
File Share
LDAP
LDAP
M
S
M
S
R
Workspace (DaaS)
VPN GW
VPC Peering
us-east-1 (Virginia)
VPN GW
VPC
Kubernetes Cluster
LB Service
管理サーバ
Amazon EKS
k8s Master
Node
Node
Node
リモートアクセスユーザ(SSL-VPN)
リモートアクセスユーザ(DaaS)
SSL
社内ユーザ
社内VPN Gateway

中級課題

IoTセンサーを用いた在席確認システムの構築

概要

- 課題の内容：
  - 単体でインターネットに直接接続可能なIoT機器を利用し、クラウドサービスと連動したサービスを構築する
- 要件：
  - センサは複数個利用すること
  - 席の着席はセンサによって自動で取得する
  - センサは何をつかってもよく，また、主題ではないため、センサによる在席情報取得の精度は問わない
  - センサデータは、サーバ（クラウド側）で収集し、データベースに保存する
  - 席の着席の状況をリアルタイムで確認できるwebシステムを作成する
  - パブリックなクラウドサービスと連携（マッシュアップ）する
    - パブリックなクラウドサービスの例(IFTTT, AWS IoT Gateway, LINE Things/Notify, など)
      例) 収集したセンサデータをトリガーとして、一部屋にX人以上着席したら自動で携帯に通知(LINE APIなど)を送る
  - データ収集にあたり、プライバシー・セキュリティに配慮、工夫をする
  - 最後の成果発表時にデモできるようにシステムを構築する

目標

オフィスの集中席の**在席確認システム**をつくる

【現在の課題 + 期待する効果】

- 課題①：座席の利用状況がアナログで管理されている
  - 座席まで行かないと空いているかわからない

➡ **効果①：Webから利用状況を確認できる**

- 課題②：座席の利用状況が把握できていない
  - 利用状況を分析することができない

➡ **効果②：利用状況を可視化して分析できる**

要件 vs 設計

- センサは複数個利用すること
- 席の着席はセンサによって自動で取得する
- センサは何をつかってもよく，また、主題ではないため、センサによる在席情報取得の精度は問わない
- センサデータは、サーバ（クラウド側）で収集し、データベースに保存する
- 席の着席の状況をリアルタイムで確認できるwebシステムを作成する
- パブリックなクラウドサービスと連携（マッシュアップ）する
  - パブリックなクラウドサービスの例(IFTTT, AWS IoT Gateway, LINE Things/Notify, など)
    例) 収集したセンサデータをトリガーとして、一部屋にX人以上着席したら自動で携帯に通知(LINE APIなど)を送る
- データ収集にあたり、プライバシー・セキュリティに配慮、工夫をする
- ~~最後の成果発表時にデモできるようにシステムを構築する~~

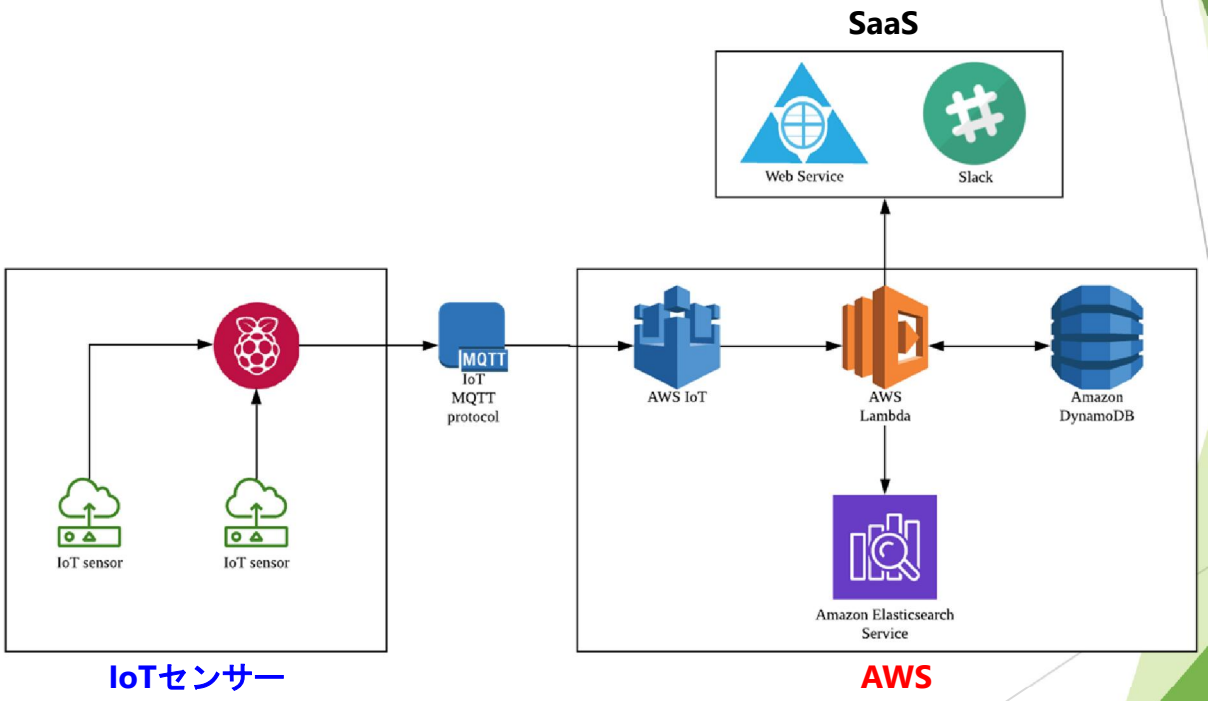
【独自に追加】

- センサ⇄クラウドの拡張性を高める
- サーバーレスで実現する
- センサーデータを可視化する

◆色分け

- IoT Sensor：青
- AWS：赤
- SaaS：緑
- Other：黒

全体構成図


SaaS
Web Service
Slack
MQTT
IoT
MQTT
protocol
AWS IoT
AWS
Lambda
Amazon
DynamoDB
IoT sensor
IoT sensor
Amazon Elasticsearch
Service
IoTセンサー
AWS

在席確認のロジック：明るさセンサー

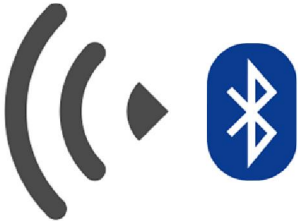
椅子の背もたれに明るさセンサーを設置

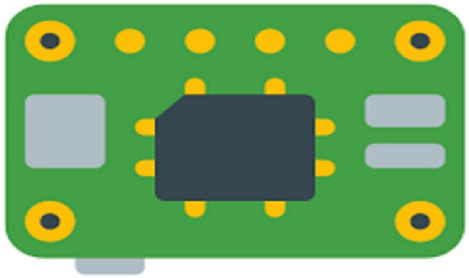
在席＝暗い

離席＝明るい

IoTセンサー

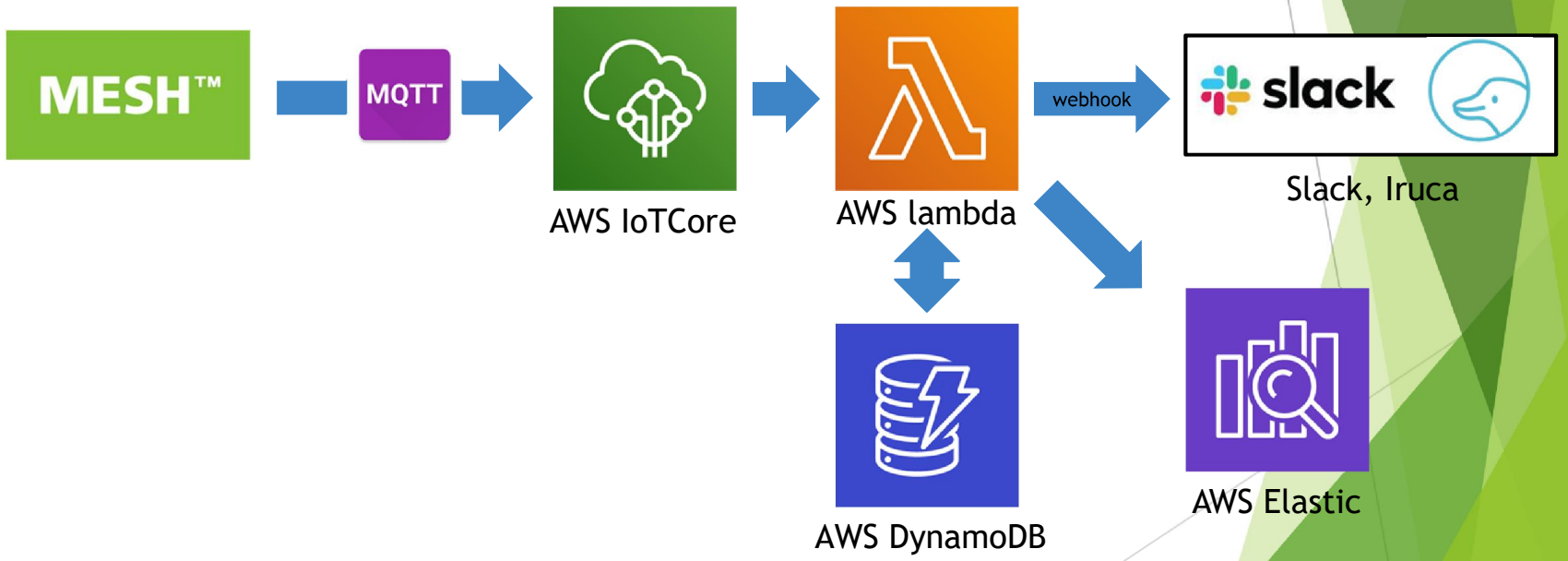
MESH
明るさセンサー



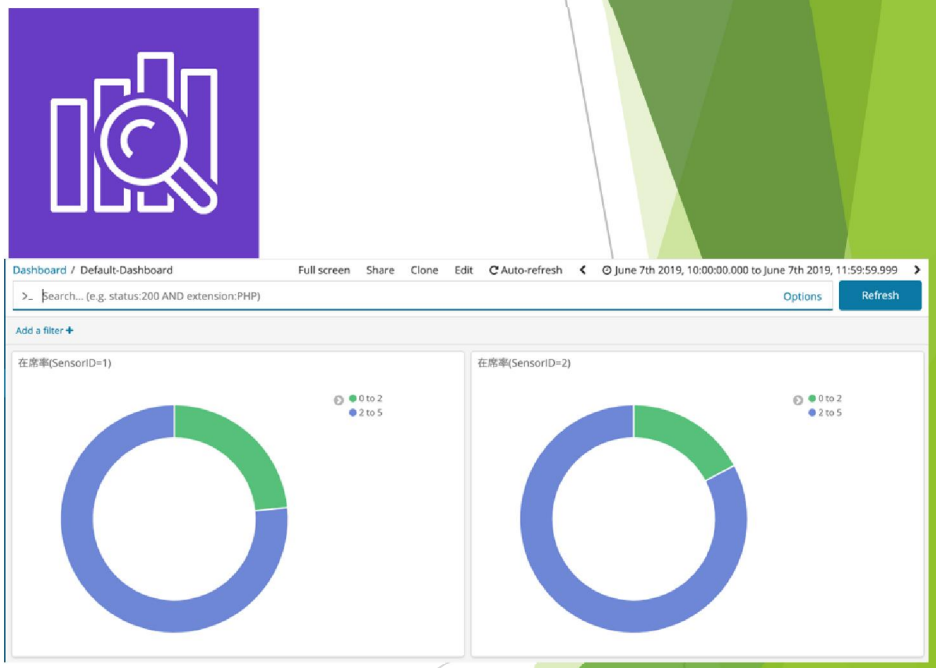


MESHハブアプリ
インストール済み
Raspberry Pi

AWS　実装


MESH™
MQTT
AWS IoTCore
AWS lambda
webhook
slack
Slack, Iruca
AWS DynamoDB
AWS Elastic

在籍確認（SaaS, 可視化）



STEP2の感想

STEP2の感想

学んだこと
- 他社のエンジニアとのコミュニケーションを取りながらのプロジェクト進行
- 実務では触れることがない技術

参加してよかったこと
- 自由度が高い
  - クラウド、IoTセンサ、NW機器、サーバーなど、リソースが自由に使える
  - 要件内であれば構成はチームごとに自由
  - 検証環境のためTry＆Errorで開発を進められる
- 要件定義から開発までチームで実践的に取り組める
  - STEP1の座学で学んだ内容を実践、知識の定着
- 成果発表時のフィードバックで実務運用者視点でのフィードバックをもらえる

改善点
- 業務と並行して研修を進めるのは大変
  - 本人のやる気と会社の理解が必要