



aws SUMMIT  
ONLINE

JAPAN | MAY 11-12, 2021

PAR-15

# インフラエンジニアだけで サービス開発してみた

～フルスタックエンジニア育成への取り組み～

広野 祐司

SCSK株式会社



## 広野 祐司 (ひろの ゆうじ)

### SCSK株式会社

- 住友商事グループのシステムインテグレータ
- AWSパートナーネットワークの最上位である  
プレミアコンサルティングパートナー



#### 👤 役割

CCoE 教育担当

#### 👤 好きなAWSサービス

AWS Amplify

#### 👤 経歴

当社DCのマネージドサービス運営

海外マネージドサービスの国内展開

当社がどのようにして  
フルスタックエンジニアを  
育成しようとしてきたか

試み

成果

成功要因

今後

## インフラエンジニアだけでサービス開発してみた

～フルスタックエンジニア育成への取り組み～



社内のインフラエンジニアを集めて、  
当社社員向けAWS学習サイトを開発した



AWS Certified Solutions  
Architect - Associate

サンプル問題集

全10問

▶ 使い方を表示

問題番号



## 問題

カスタマーリレーションシップマネジメント (CRM) アプリケーションは、アプリケーションロードバランサーの背後にある複数のアベイラビリティゾーンの Amazon EC2 インスタンスで実行されます。

これらのインスタンスの 1 に障害が発生した場合、どうなりますか？

- A) ロードバランサーが、障害が発生したインスタンスへのリクエストの送信を停止する。
- B) ロードバランサーが、障害が発生したインスタンスを終了する。
- C) ロードバランサーが、障害が発生したインスタンスを自動的に置換する。
- D) ロードバランサーが、インスタンスが置換されるまで、504 ゲートウェイエラーを返す。

## 解答・解説

▼ 解答・解説を表示

A - アプリケーションロードバランサー (ALB) は、正常なインスタンスにのみリクエストを送信します。ALB は、ターゲットグループ内のターゲットに対して定期的なヘルスチェックを実行します。設定可能な回数だけ連続してヘルスチェックに不合格だったインスタンスは、正常ではないと見なされます。ロードバランサーは、次のヘルスチェックに合格するまで、インスタンスにリクエストを送信しなくなります。

**AWS認定の試験サンプル問題を掲載。  
スマホ対応。  
スキマ時間に勉強できる！**



## AWS Certified Solutions Architect - Associate

Step1 リージョンとAZ (4:01)



## セミナー資料



### 試験対策勉強会資料

2020.8.6

全社向け研修で使用した資料  
です。2020年3月23日より

リニューアルされた試験 SAA-C02 向けに改訂  
されています。

自作した学習用動画コンテンツや、  
社内研修テキストも掲載。

# 【余談】 Hirodemy の姉妹サービス「ひろのくん」

HIRONOKUN 2020

- ホーム
- Dashboard
- AWS認定試験 <
- Cloud認定試験 <
- ITM認定試験 <

ひろのくん 2020 Education Statistics and more



ログイン



当サイトは、ITM部門 IaCカリキュラム/AWS認定資格の

## Dashboard

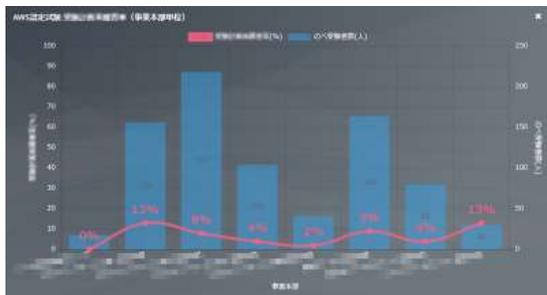
### AWS認定試験 ノミネート・進捗状況サマリー

試験名	申込率	合格率	ノミネート者数	申込者数	合格者数
CP(Cloud Practitioner)	100%	89%	88	88	48
SA-A	100%	54%	440	438	172
SA-P	100%	54%	160	160	50
Dev-A	100%	90%			
Dev-P	100%	95%			

申込率: 申込者数 / ノミネート者数 \* 100  
合格率: 合格者数 / 受験者数 \* 100

### SA-P試験

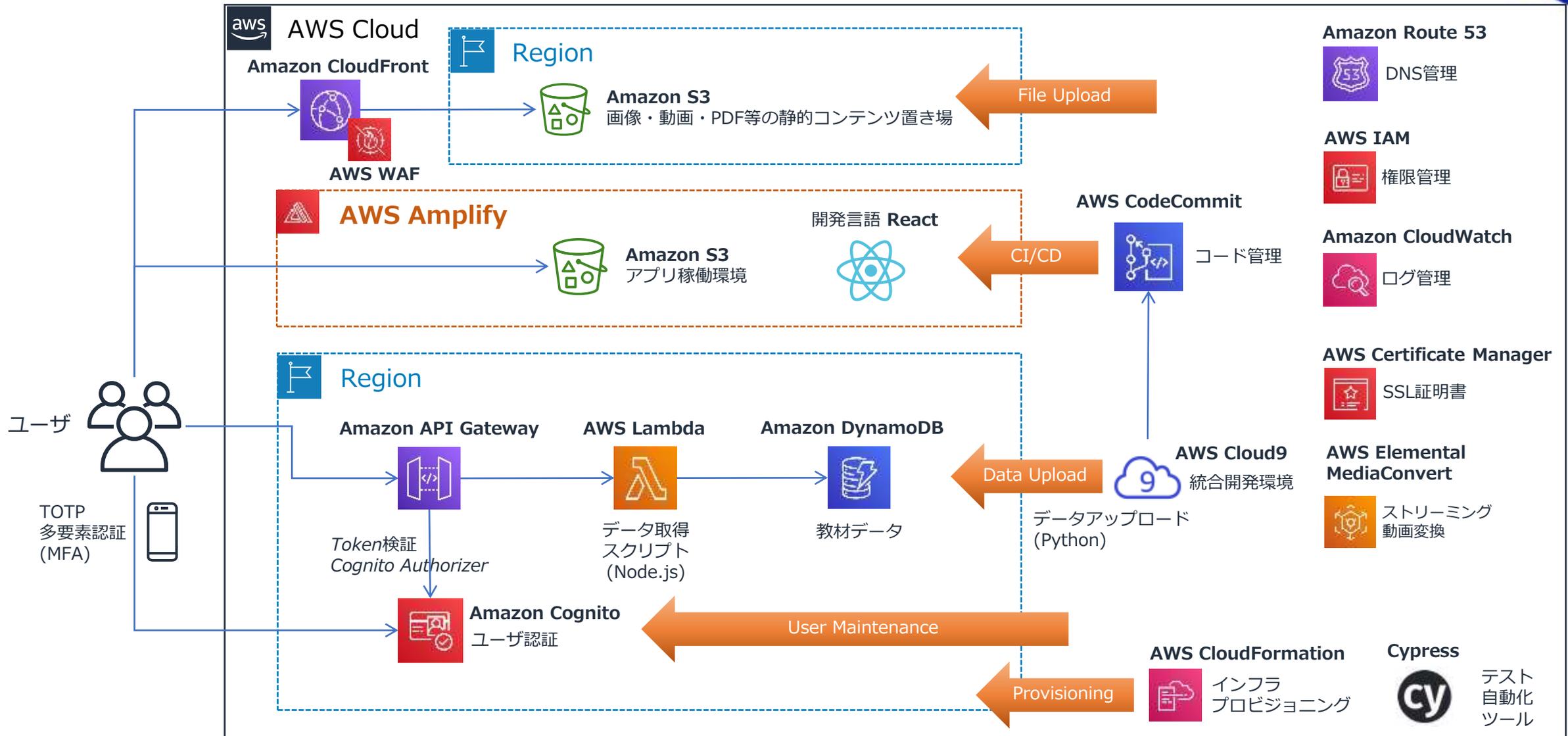
部署名	受験計画率	申込率	合格率	ノミネート者数	受験計画者数	申込者数	合格者数
ITM部門	100%	100%	25%	4	4	4	1
システムセンター	100%	100%	41%	38	38	38	7
システムセンター	100%	100%	80%	25	25	25	12
システムセンター	100%	100%	70%	17	17	17	7
システムセンター	100%	100%	75%	4	4	4	3
システムセンター	100%	100%	50%	51	51	51	16
システムセンター	100%	100%	29%	16	16	16	2
システムセンター	100%	100%	67%	5	5	5	2
システムセンター	100%	100%	54%	160	160	160	50



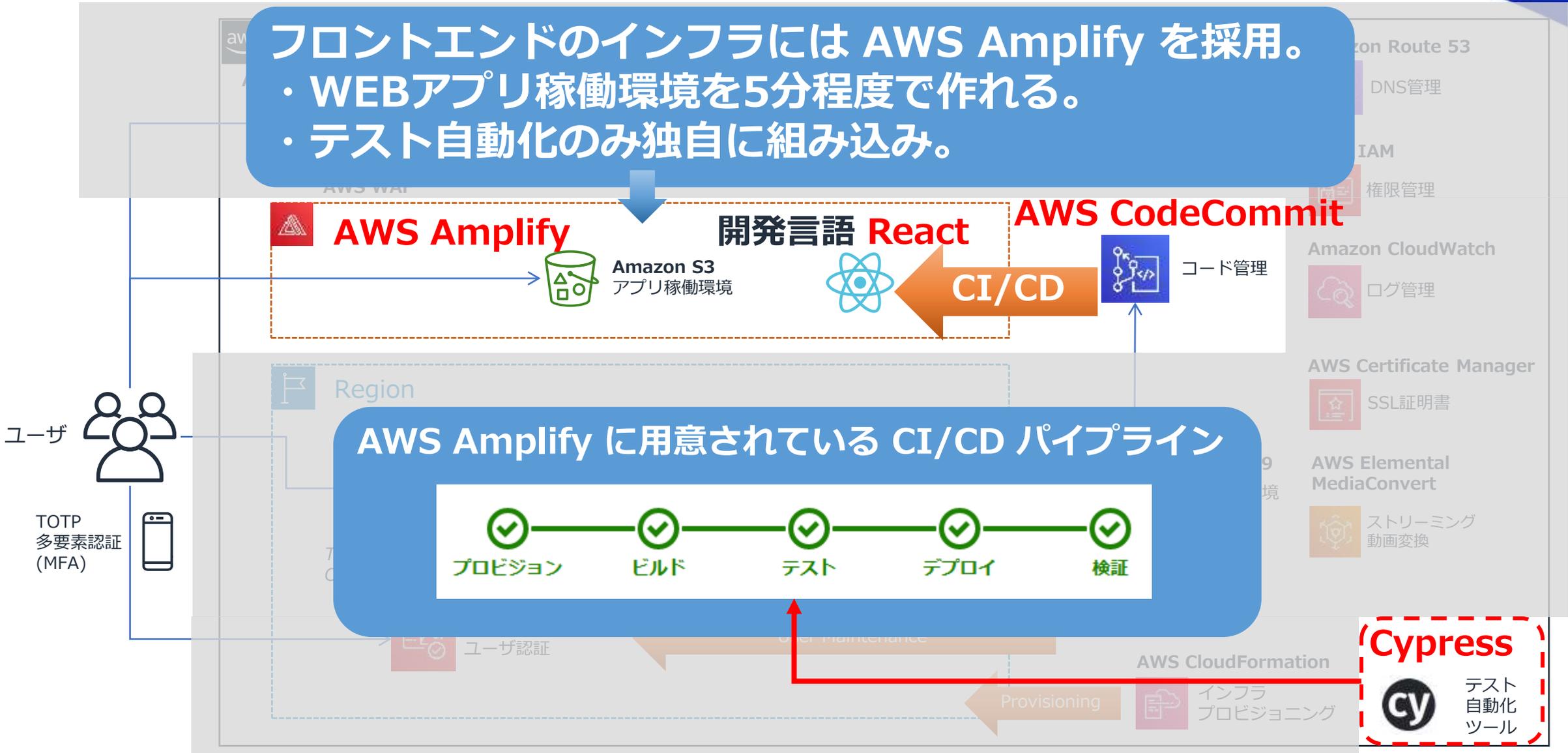
社員のAWS認定の受験状況を経営層、ライン職とリアルタイムに共有！ Hirodemyとの相乗効果で効果大！

- AWS Amplify + React による  
**モダンWEBアプリケーション**
- **AWS サーバーレスソリューション**  
のベストプラクティス集

# Hirodemy アーキテクチャ図 (全体像)



# Hirodemy アーキテクチャ図 (拡大1)

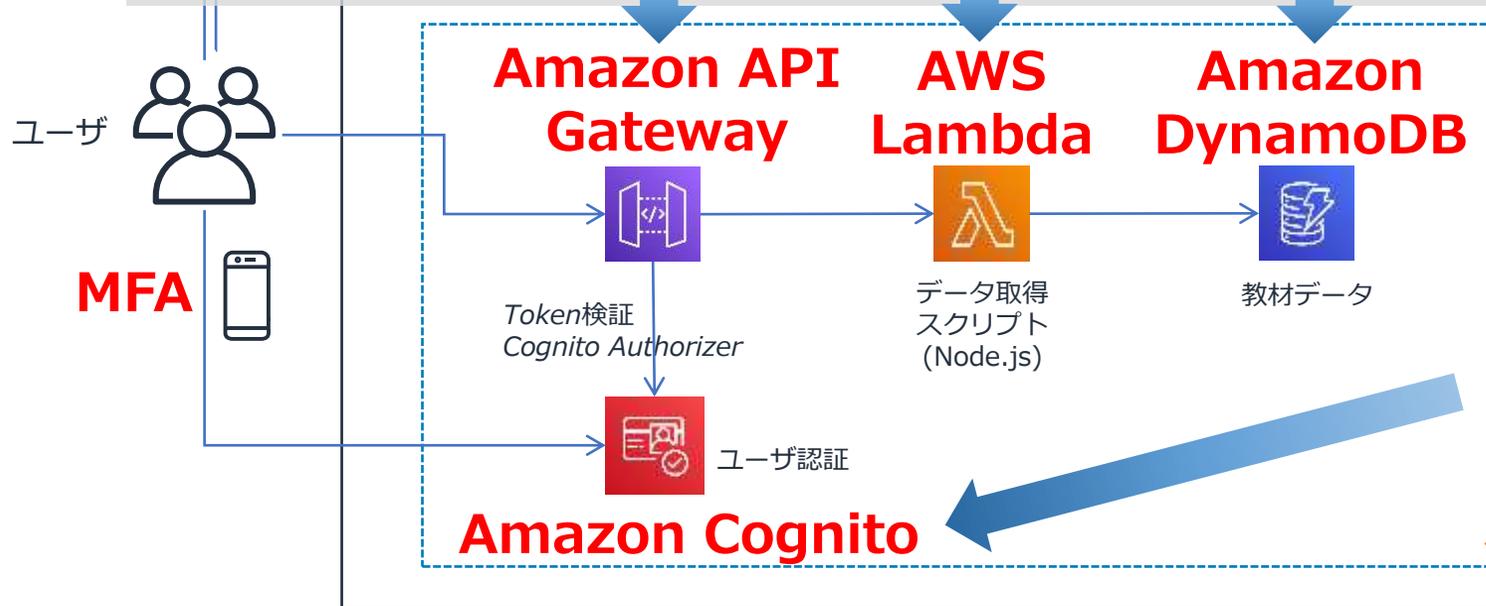


# Hirodemy アーキテクチャ図 (拡大2)

aws AWS Cloud Amazon Route 53

バックエンドのインフラは AWS サーバーレスソリューションの王道。

- ・この組み合わせはAWS認定試験の勉強をしていると必ず登場する。
- ・複雑なデータでなければこのアーキテクチャだけでだいたいのことは実現できる。

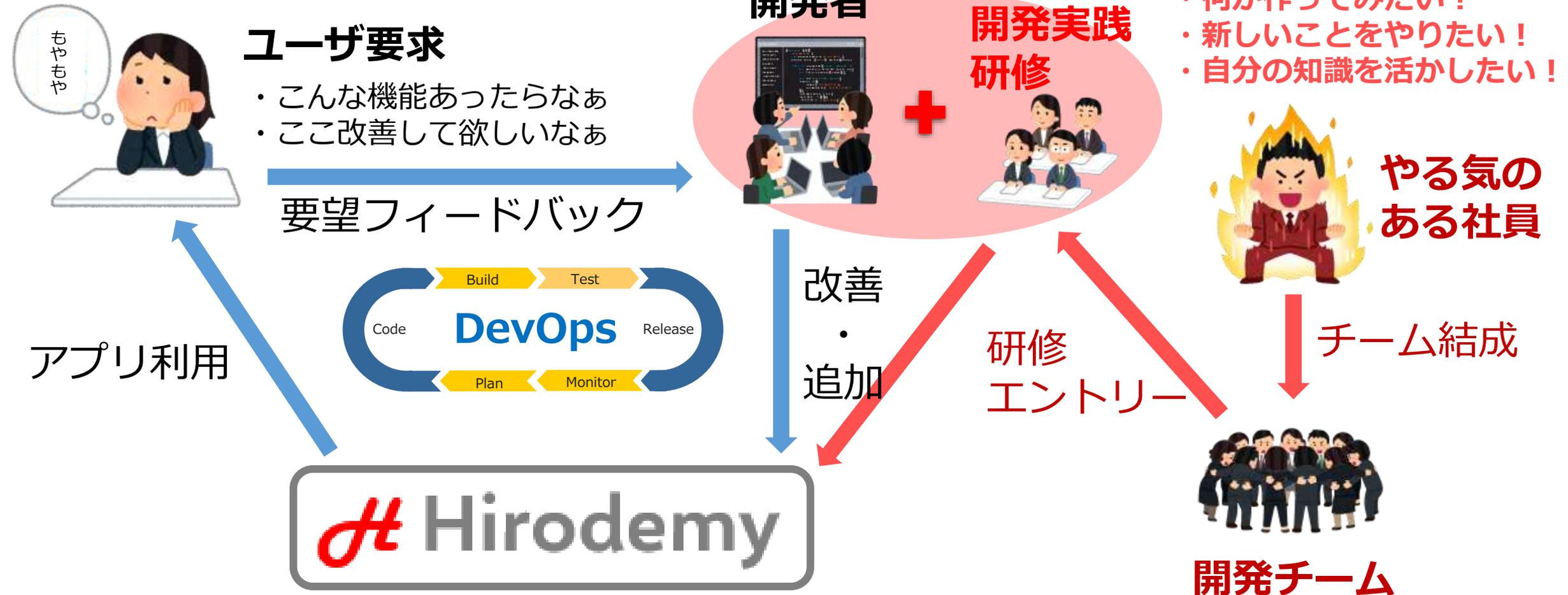


認証基盤は  
Amazon Cognito

- ・AWSドキュメント通りにコードを書くだけでログイン画面ができる！
- ・MFAも設定するだけ！



## みんなで作る教育サービス！



# サービス開発実践研修 「Endeav Dojo」

えん でぶ どうじょう

- 経験・年齢不問。必要なのはやる気のみ！
- 6名以内で開発チームを結成、エントリー！
- 2か月でサービス開発をやってみよう！



これまでインフラエンジニアを対象に開催、合計7チームが参加

## AWS AppSyncによるユーザー間チャット機能

### 問題

分析会社は、ユーザーにサイト分析サービスを提供する予定です。このサービスでは、ユーザーの Web ページに、同社の Amazon S3 パケットに対して認証済み GET リクエストを行う JavaScript スクリプトが含まれている必要があります。

スクリプトを正常に実行するため、ソリューションアーキテクトが行うべきことは何ですか？

- A) S3 パケットでクロスオリジンリソース共有 (CORS) を有効にする。
- B) S3 パケットで S3 バージョニングを有効にする。
- C) ユーザーにスクリプトの署名付き URL を提供する。
- D) パブリック実行権限を許可するようパケットポリシーを設定する。

### 解答・解説

▼  解答・解説を表示

A - Web ブラウザが、Web ページとは異なるドメイン名を持つサーバーから作成されたスクリプトの実行をブロックします。[Amazon S3 は CORS を使用して設定し](#)、スクリプトの実行を許可する HTTP ヘッダーを送信できます。

### コメント

Enter your comment

運営へこのメッセージをメールする

解答・解説の下に、  
コメント欄を追加開発



ありがとうございます！わかりやすかったです！ ❤️



このサイトがわかりやすく良いと思いますよ。

<https://xxx.xxxxx.xxx> ❤️ 2

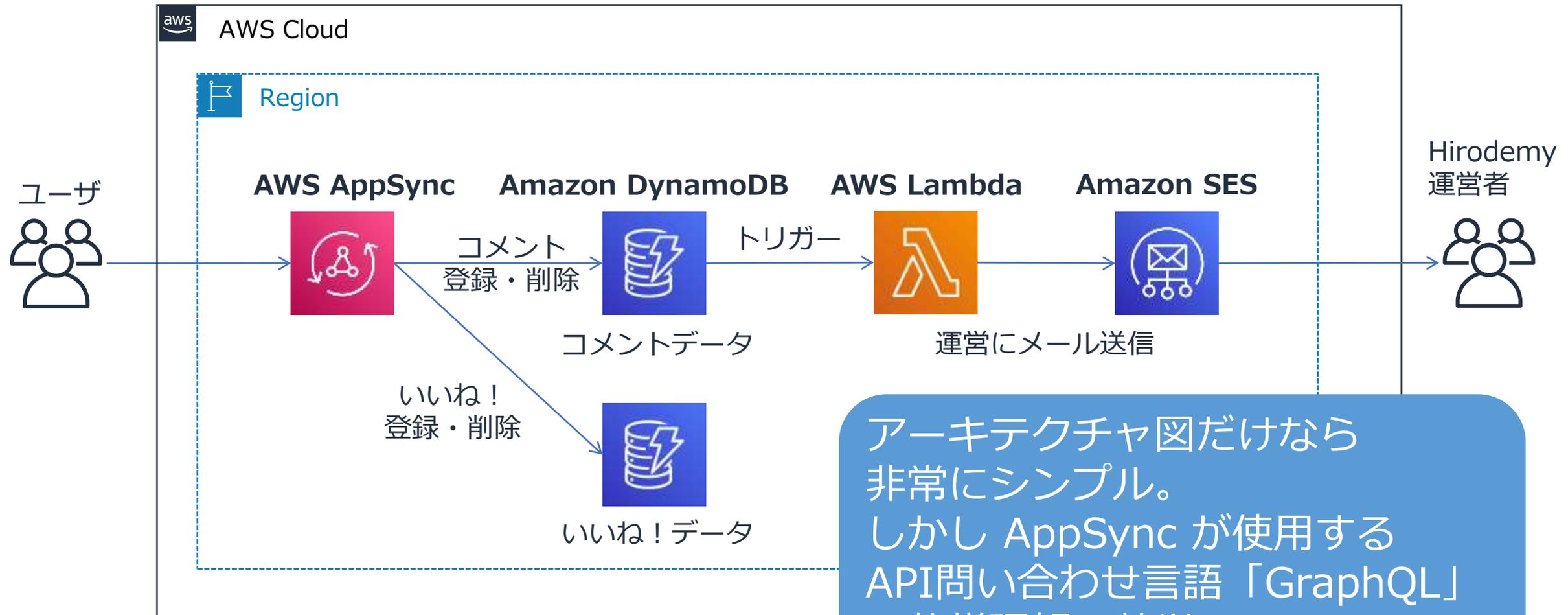
いいね！ボタン

CORSって調べてみると複雑でわかりにくいんですけど、何かわかりやすい学習サイトってないでしょうか。



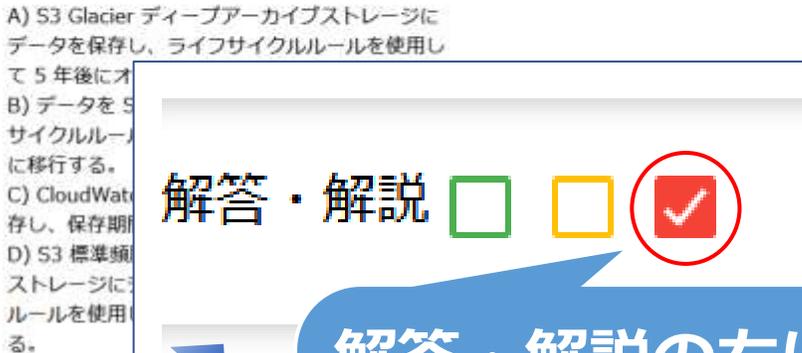
# Endeav Dojo 開発事例 1 アーキテクチャ図

## AWS AppSyncによるユーザー間チャット機能

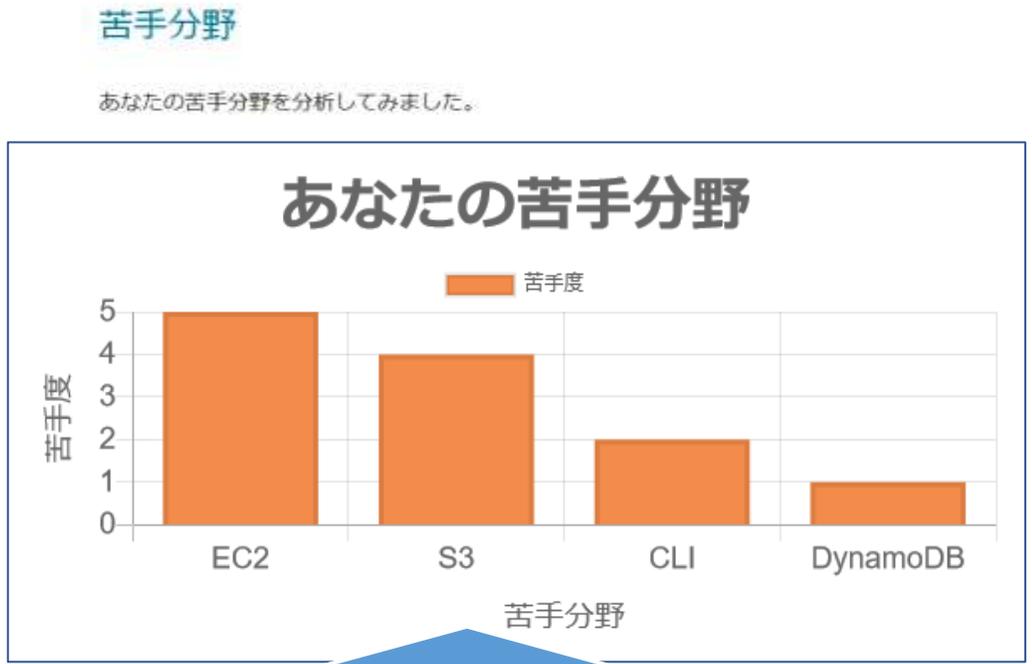


アーキテクチャ図だけなら非常にシンプル。しかし AppSync が使用する API 問い合わせ言語「GraphQL」の仕様理解に苦労。

## Amazon Comprehendによるユーザの苦手分野分析機能

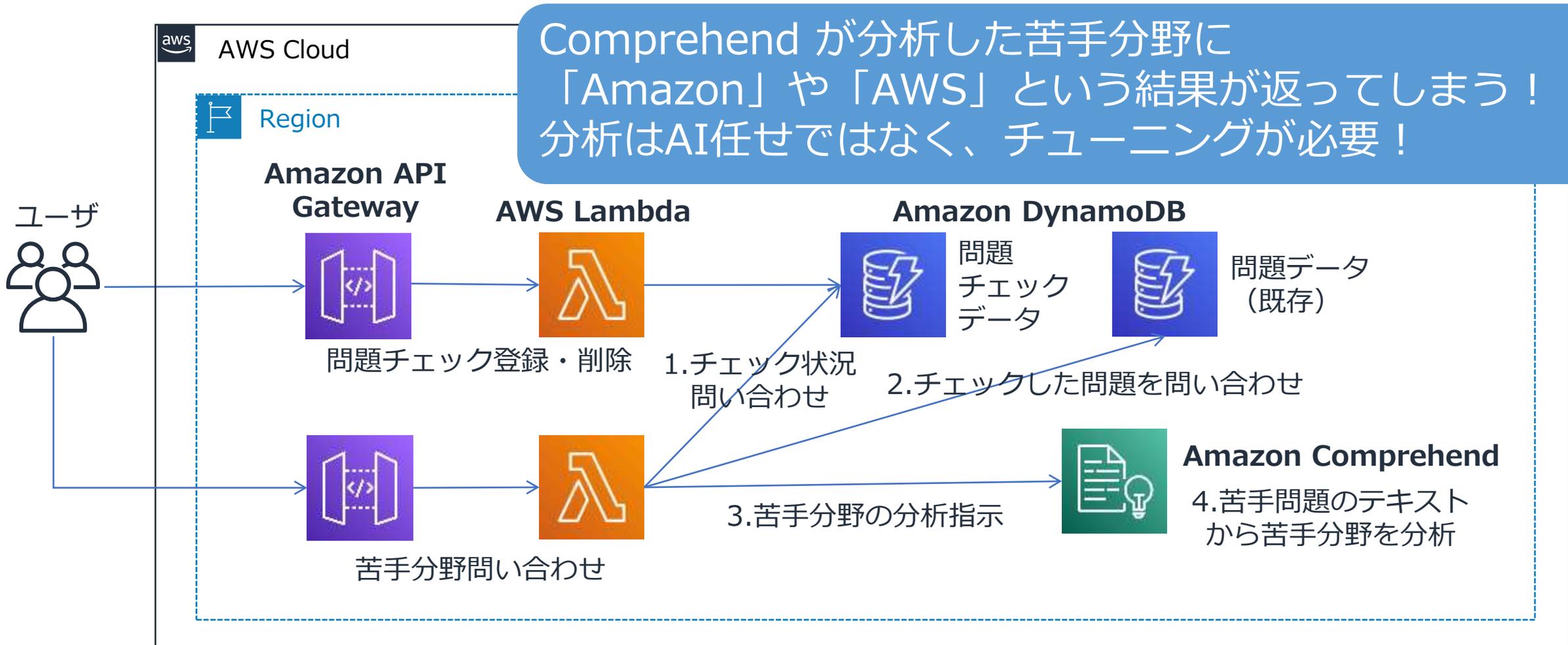


解答・解説の右に、  
チェック欄を追加。  
苦手な問題は  
**赤チェック!**



赤チェックを入れた問題の傾向から、AIが苦手分野を分析!

## Amazon Comprehendによるユーザの苦手分野分析機能



**なぜインフラエンジニアが短期間で  
サービスを開発・リリースできたのか？**

1. 設計・開発フェーズで要件と開発物の乖離がなく、手戻りが発生しなかった。
2. クラウドネイティブなサービスを利用することで、開発にかかる労力を削減できた。
3. 自己学習モチベーションを維持できた。



## 設計・開発フェーズで要件と開発物の乖離がなく、手戻りが発生しなかった。

- 研修の題材（Hirodemy）が身近なものだったため、受講者が開発物をイメージしやすかった。
- Amazon の顧客視点のサービスデザイン方法である **Working Backwards** を参考に、サービスが提供する価値（=開発要件）を整理。設計・開発フェーズの定期レビューで価値を意識する機会を設けた。



## Working Backwards :

- あなたの**お客様**は誰ですか？
- お客様の抱える**課題**や新しい**チャンス**は何ですか？
- お客様にとっての**メリット**は明確になってますか？
- どのようにして**お客様のニーズ**を知ったのですか？
- お客様はどのような**顧客体験**を得られますか？

お客様を起点に、サービスのプレスリリースを書いてみる！

## サービス要約

### 「理解度チェック機能」オープン

これまで苦手な試験問題を感覚的に管理してきた方へ  
Hirodemy上で苦手な問題分野が可視化されるようになりました！

## お客様の声

この機能を利用することで過去に自分がどの問題を苦手と感じていたか客観的にわかるようになり、試験直前に復習すべき問題が明確になりました。おかげで、一回で試験に合格することができました。ありがとうございました。

**開発中、迷いが出たらプレスリリースを振り返る！**

Working Backwards で「つくるもの」は決められた。

でもどうやってつくればいいのか、頭は白紙状態！



クラウドネイティブにつくろう！

AWSのマネージドサービス、ベストプラクティスを調べ、フル活用しよう！

実現できそうなアーキテクチャを検討し、AWS ソリューションアーキテクト様とレビュー！

非機能要件にかかる労力を最小限に。  
クリエイティブな部分に注力できる。

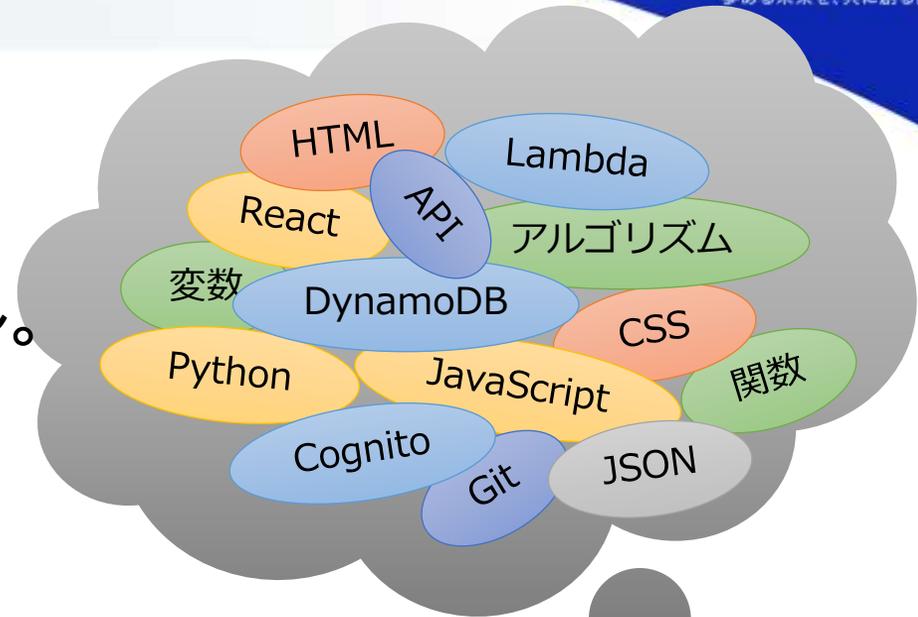
いざ、開発フェーズへ。

基本、自己学習※が Endeav Dojo のスタイル。

- わからないことがいっぱい。
- 何から手を付けていいかわからない。
- うまくいかない、逃げ出したい・・・。



なぜ逃げ出さなかったのか！？  
それが成功要因。



開発フェーズの成功要因は、自己学習モチベーションの維持。

なぜモチベーションを維持できたのか？



おもしろい！

楽しい！



この気持ちが、受講者の根底にあった！

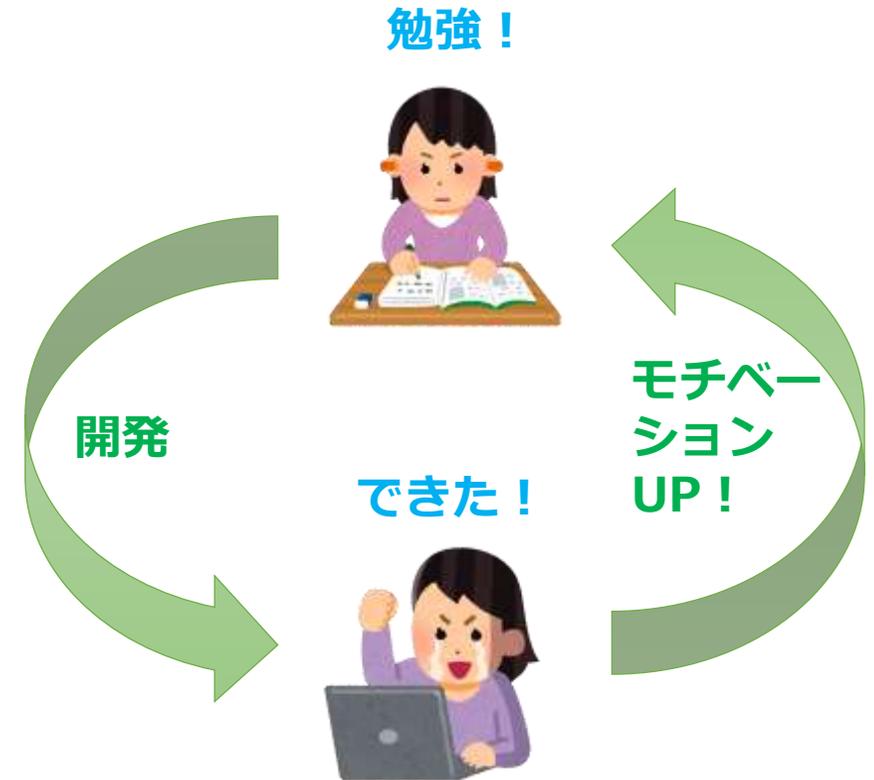
## なぜ自己学習を楽しめたのか？

「できなかったことができるようになる」

「つくったもの（ユーザに届ける価値）が目に見える」

できるところから少しずつ開発 ※

その積み重ねが自信になっていく



そして、ひとりじゃなかった。

## Endeav Dojo コミュニティ

困っても  
メンバーで  
助け合える



いつも  
見守って  
いるよ



俺も頑張らなきゃ！



基本は自己学習



頑張っている人  
は周りにも好影響  
を及ぼす

行き詰まっても  
AWSソリューション  
アーキテクト様や  
研修運営スタッフが  
アドバイス



1. 設計・開発フェーズで要件と開発物の乖離がなく、手戻りが発生しなかった。
2. クラウドネイティブなサービスを利用することで、開発にかかる労力を削減できた。
3. 自己学習モチベーションを維持できた。



仕組み的な要素はあったにしても、  
「おもしろい」「楽しい」と思ってもらえたことが  
リリースまで到達できた一番の原動力

## ■■■ 学びとなったこと ■■■

- 研修で利用したAWSサービスやコーディング関連の知識だけでなく、実際に手を動かすことでより理解が深まるなと感じました。今までは本やネットで調べて読むだけが多かったのですが、実際に触って動かしてみることによって理解が進むことが実感できたので、今後も手を動かしてみようことを忘れずにいたいと思いました。また、今後は開発のスピードも早まっていく中で、新しい知識をどんどん吸収し、挑戦していくことがとても重要であると思いました。インフラエンジニアに求められる開発スキルの必要性など、今後のキャリアにも大きく関わる内容が得られて素晴らしかったです。
- 普段業務でインフラ側に従事しており、フロントエンドとバックエンドのつながりの部分を考えてことがありませんでした。そのためフロントエンド→バックエンドのデータの流れやネットワークを理解する良い機会になりました。また普段プログラミングをしないので、試行錯誤して実装できた機能には愛着がわきました。
- DevOpsの必要性、CI/CDの利便性等、インフラでの仕事ではあまり関わらない点に目を向けられた点、ウェブアプリ開発を学びました。
- 時代はサーバーレスということをもっと体感することができました。
- クラウドサービスのつながりを実装の中で確かめていけるのは面白かったです。座学でつながることは分かっていても技術的にどう設定しているのかは分からないのでとても学びになりました。
- プレスリリース～開発～成果発表まで実業務に活かせるような内容で非常に勉強になりました。ぜひ来期も継続して開催をお願いします。

## ■■■ モチベーション ■■■

- 元々アプリの知見がほとんどなかったため難しく感じましたが、やっつけて楽しいという気持ちを頼りに乗り切りました！
- 最初はReactが難しかったですが、みんなで乗り越えることができました。
- チームでひとつのものを作り上げられたときの達成感！
- フロントエンド・バックエンドで構築したそれぞれの仕組みが、一つになって機能した時にとっても達成感がありました。
- 実際に作ったものが目に見える上に、ちゃんと動いた時の達成感が感じられて楽しかったです。
- 本当に使ってもらえるサービスになれるように注力しました。
- チームメンバーの方の熱意や技術力にたくさん刺激を受けたので、自分にとっていい収穫となりました。
- 今回、インフラしか見れていなかった自分の世界が広がったと感じています。おかげ様で、今後開発についてもアンテナ張って色々試してみようと思えるようになり、今後のステップアップに弾みがつくようになりました。
- IT業界は今後大きく変化していくと思います。その中において新しい知識はがむしゃらに取得しないといけないと思いました。
- 今後、AWSを業務で使用していきたいと切に思いました。

## ■ ■ ■ 研修の進め方 ■ ■ ■

- 自分たちで題材を決めて実装していくので、他の研修よりも理解できました。
- プレスリリースに記載した内容を実現することに注力して進めることができたと思います。そして機能改修、ブラッシュアップという流れは普段のプロジェクトでは体感できないフローだったので大変興味深かったです。
- 開発をしている中で、チーム内で意見が発散しかけた際に改めてプレスリリースを意識することで話を収束させることができました。
- 実際に画面を作成した後、レイアウトや表現する文字など、利用者視点で気になる点も意識しました。
- チームでコミュニケーションを取り、互いに不足分を補いながら進めていけたのはいい経験になりました。
- 研修運営スタッフの方に、わからない箇所のモヤモヤが「スッと」するように的確に助言頂けたため、ストレスなく受講することが出来ました。（もし、モヤモヤが続いていたらAWS拒否反応が強くなっていたと思いますので、良かったです！）しかも直接の質問でない箇所にも助言があったりと、頻繁に各チームのSlackを見て頂いている安心感もありました。言い換えれば、OJT指導員みたいな感じでしょうか。受講者に寄り添う姿勢が心地よかったです。ありがとうございました。
- 時間を自由に使える点が非常に良いと感じました。小さなプロジェクトが動いている感じで、自発的に動けることで、他の研修とは一線を画していて良かったです。開発期間が短く、引き締まった研修で非常に勉強になりました。

# インフラエンジニアだけでサービス開発してみた

～フルスタックエンジニア育成への取り組み～

- Endeav Dojoをさらに改善、社内開発者向けに展開
- エンドユーザ様向け内製化支援サービスを提供中
- AI系の教育も検討中



# Thank you!

広野 祐司

インフラエンジニアだけでサービス開発してみた  
～フルスタックエンジニア育成への取り組み～