

iOSアプリエンジニアが AWSreInventに行ってきた

(真面目編)

JAWS-UG大阪
2025.01.16

とと (@totokit4)



とと (平口 咲)

X: @totokit4

- 南海電気鉄道で働いています
- iOSアプリエンジニア
- AWS re:Inventは初参加
- SNSに写真投稿ご自由にどうぞ



iOSに
AWSが

何の関係があるんや

でも会社が行かせてくれることになったから行っとう

iOSアプリエンジニアが
Apple, Mac, Mobile で
検索して
見つけてきたセッション5選



DEV334_Swift server side serverless

Swift、サーバーサイド、サーバーレス

- iOSアプリの開発に必要な不可欠な言語、Swift
- Swiftの強み: **高速、安全、モダンな構文!**
 - **サーバーサイドにもピッタリですよね??**
- 構築の流れ:
 1. OpenAPI仕様作成: API定義を標準化。
 2. 依存関係の設定: Package.swiftで必要なライブラリを指定。
 3. ビジネスロジックの記述: Swiftでサーバーロジックを実装。
 4. AWS Lambdaへのデプロイ: AWS SAMを使用して効率的にデプロイ。
- AWSサービスの統合: REST API Gatewayやスケジュールトリガーでサーバーレス運用。
- **利点: フロントエンドとサーバーサイドを同じ言語で書ける!** Appleエコシステムとの連携できる



CMP328_Apple Vision Pro apps on AWS Cloud

AWS クラウド上の Apple Vision Pro アプリ

- Spatial Computing: 3D、動作、時間・位置データを統合
- ユースケース: デジタルツイン、VRショールーム、予知保全、リモートオペレーションなど
- AWSサービスの活用:
 - **AWS Swift SDK: Vision Pro向けアプリ開発にも使える**
 - Amplify: ユーザー管理、データベース構築、通知機能をサポート
 - Amazon S3: データのアップロード/ダウンロード
- 生成AI: Amazon SageMakerやBedrockでAIモデルを管理・運用。
- 学習リソース: AWS Skill BuilderやRamp-Up Guidesを活用して知識を拡張。

AWS Solutions Library / Guidance

Guidance for Developing Apple iOS and Vision Pro Applications with Unity on Amazon EC2

FWM204_Optimizing the worlds top apps How Meta tests using AWS Device Farm

世界トップクラスのアプリの最適化: AWS Device Farm を使用した Meta テストの方法

- **モバイルテストの課題:** デバイスの多様性、デバイスを揃えるコスト、接続性、端末機能のばらつきの問題。
- **AWS Device Farmの特徴:**
 - 実デバイスを使用した自動テストやリモートアクセスができる
 - カスタムスクリプト対応で高度なテストが可能
- **Metaのテスト手法:**
 - ノイズ削減（温度管理、状態リセット、プロキシ活用）
 - ルート化デバイスで精度向上
- **新機能:**
 - XCUIテストプラン対応、リモートアクセスとエラー解析の改善、33%高速化。

CMP410_Accelerate testing cycles of CI CD pipelines with EC2 Mac instances

EC2 Mac インスタンスを使用して CI/CD パイプラインのテスト サイクルを加速する

- Appleアプリの開発にはMacOSとXcodeが必須
 - 加えて、オンプレミス管理の負担、スケーラビリティのなさなどが課題
- EC2 MacでmacOS環境を用意（数分で構築できる！）
- 成果: ベータ版macOSの早期採用、 Xcode更新速度4倍向上、テスト失敗80%減少
リリースまでの時間短縮
- 更新情報（2024年）：
 - M1 Ultra Mac Studio対応、新しいmacOSサポート、GUI接続性の改善

CMP210_Modernize Apple platform development with AWS and EC2 Mac

AWS と EC2 Mac で Apple プラットフォーム開発を近代化

- Apple Silicon搭載のEC2 Macインスタンスで大規模iOS開発を最適化。
 - ベアメタルスナップショットでインフラ管理を簡素化。
 - Auto Scalingグループで柔軟にスケール調整。
 - HashiCorp PackerでAMIを効率的に構築。
- 成果：シミュレータ起動時間を170秒から36秒に短縮、オンデマンドでスケーラブルなCI環境を実現。
- 推奨事項: キャッシュのクリア、EBSの最適化、AZ間でのリソース計画。



まとめ

- わいインフラ詳しくないなと自覚が深まった
- フロントって作ったものが分かりやすい=楽しい と思ってたけどインフラもおもしろいかも？
- Swiftでサーバサイド書けちゃうんだ～へ～（やるしか？）
- AWSでiOSアプリのCI/CD構築できるんだ～へ～（やるしか？）



アラサー女子が行く！ラスベガス出張



とと 1か月前



11



ととが行く！AWS re:Invent2024



とと 1か月前



10

