



Tech Post Cast

人気のIT技術記事をAIが解説する
ラジオ番組を毎日配信

Webサービス開発者の会 #17

2025.03.19

Sumihiro Kagawa

目次

- 自己紹介
- Tech Post Cast とは
- 開発した経緯や背景
- DEMO
- システム構成
- こだわりポイント
- AI ラジオ番組生成フロー
- 開発スケジュール
- 運用コスト

sumihiro3 の自己紹介



- 名前
 - 加川澄廣
- 居住地
 - 兵庫県
- 所属
 - 株式会社ブレイブテクノロジー
 - LINE ミニアプリの順番待ちサービスなどを開発
 - 取締役、開発責任者
- その他
 - LINEヤフー社認定 LINE API Expert (2019年～)

略歴

いつ頃	何をしていたか
大学生	商学部経営学科
新卒	簿記の資格を活かしたく在阪企業の経理部で財務・経理に従事した
27歳	プログラマーへ転職し東京へ
～35歳	請負開発やSESで開発、設計、リーダー、アーキテクト等を担当した
～48歳	ITコンサルで主にPMを担当した（製造業、電子書籍サービス、自治体など）
43歳頃	関西に戻った後にLINE APIでの開発やハッカソン、個人開発、ハンズオン活動に目覚める その活動がきっかけでLINE API Expertに認定いただく
48歳～	LINE ミニアプリを使った順番待ちサービスの開発責任者（現職）
2025年2月	Tech Post Cast を個人開発してリリース

Tech Post Cast とは

Tech Post Cast とは

人気IT技術記事をAIのMCが解説する AI ラジオ番組配信サービス

- 最新技術トレンドをキャッチアップできるよう、Qiita の人気IT技術記事をAIのMCが解説するラジオ番組を毎日配信しているサービス



Tech Post Cast とは

主な機能

- 毎朝7時に自動で番組を生成して配信
- Qiita人気記事を AI のMC「ポステル」が解説
- リスナーからのお便りを紹介

開発した経緯や背景

- 日々進化する技術トレンドを簡単にキャッチアップしたい
- 散歩や家事などの「ながら時間」活用したい
- 習慣化できるように毎日決まった時間に生成・配信してほしい

DEMO

Demo Page

 Tech Post Cast

2025-2-28 のヘッドラインピック

GitHub Copilot AgentとLinuxディストリビューションなど

▶ 1:36 / 6:45

🔊 ⋮

紹介記事

チャプター

番組の台本

1. オープニング

2. イントロ

3. 紹介記事 1: 「Clineに全部勝てる」勇気がでないのでGitHub Copilot Agentでお安く試してみる

4. 紹介記事 2: Oracle Exadata Database Service: 複数スタンバイ Data Guard 構成してみてみた

5. 紹介記事 3: 「Googleスプレッドシート × Gemini AI」でもう手作業はいらない！業務効率化の最強テクニック

6. 紹介記事 4: iOSアプリエンジニア（および採用者）のためのMacの選定基準

7. 紹介記事 5: わけがわからないLinuxとディストリビューションまとめ

8. エンディング

よく似た番組

2025-3-2 のヘッドラインピック

ターミナルツールとPower Appsの魅力など

2025-3-16 のヘッドラインピック

AIで病理画像分類とシェアしたくなるWeb小物など

2025-1-13 のヘッドラインピック

Pythonバリデーション法とCline開発法など

前後の番組

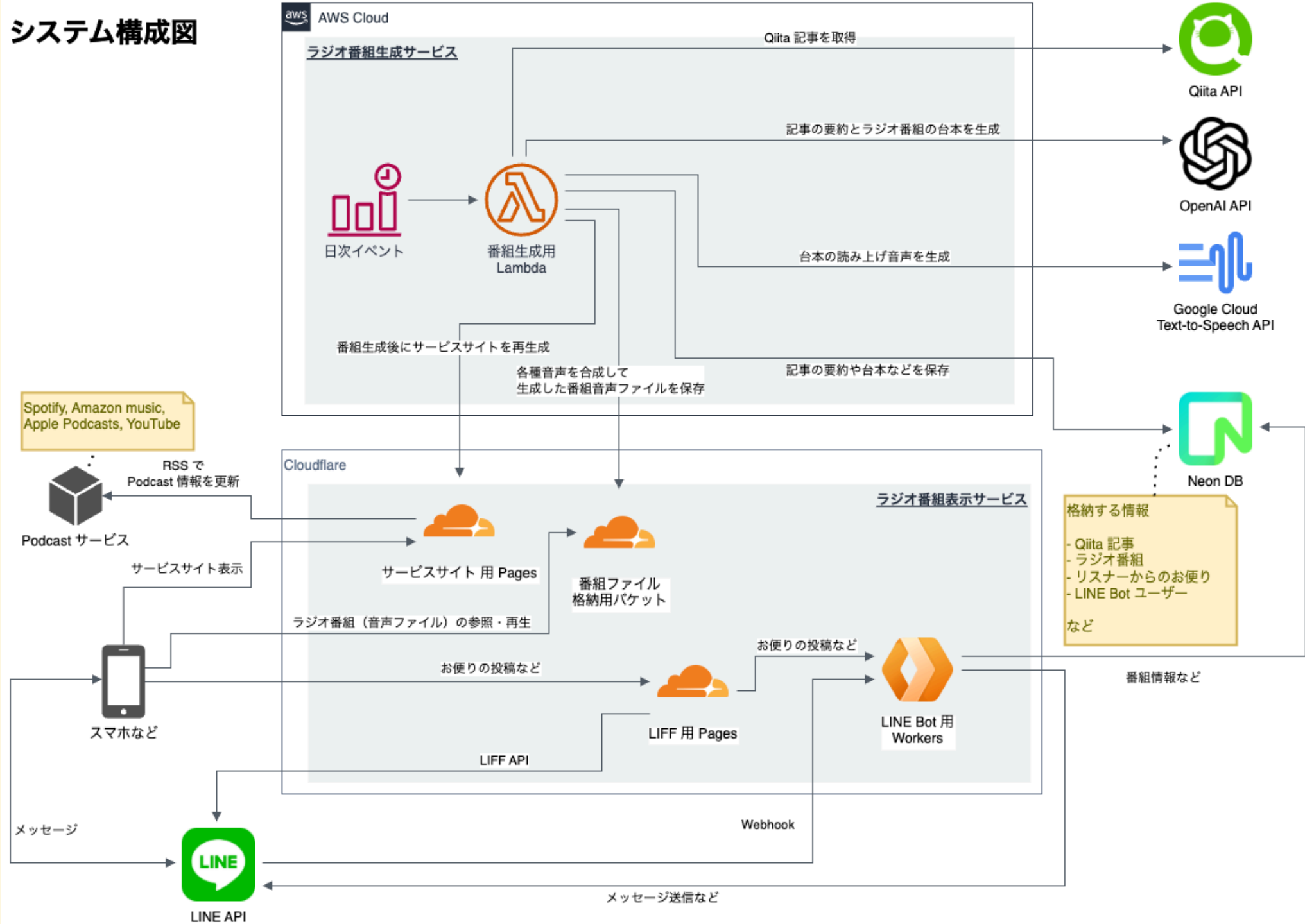
2025-3-1 のヘッドラインピック

2025-2-27 のヘッドラインピック

9

システム構成

システム構成図



ソフトウェア構成

フロントエンド

- Nuxt3 (SSG)
 - サービスサイト
- Nuxt3 (SPA)
 - リスナー投稿フォーム
- Vuetify
- TypeScript

バックエンド

- NestJS (番組生成用バックエンド)
- Hono (LINE Bot 用バックエンド)
- Prisma
- TypeScript
- FFmpeg
 - 音声ファイル編集・合成
- AWS Lambda Web Adapter

こだわりポイント

こだわりポイント（1/n）

本物のラジオ番組感を出す

- パーソナリティの個性と自然な話し方
 - 番組台本生成プロンプトで、MC「ポステル」の性格や口調などを設定
 - Google TTS で、声の高さや話す速度をチューニング
- 定型的なオープニング・エンディング構成
 - 実際のラジオ番組のように、オープニングで MCの挨拶や日付に応じたコメントを挿入
 - エンディングでは、紹介した記事を簡単に振り返る
- ユーザーエンゲージメント
 - リスナーからのお便りを紹介して、MC がコメントする

こだわりポイント (2/n)

番組の聴きやすさを重視

- **AIが対応していない最新の用語を補正**
 - 間違った読み方をしている場合は用語登録して補正する
- **BGMや効果音を活用**
 - 音声合成処理とは別で BGM や、記事間の効果音を合成してメリハリをつける
- **チャプターですぐに聞きたい箇所へ**
 - オープニングや各記事、エンディングなど好きな箇所へワンタップで到達できる

こだわりポイント (3/n)

定常処理の完全自動化

- 番組生成や Podcast サービスへの配信といった定常処理は完全自動化
 - 毎朝定時に番組の音声ファイルを生成開始
 - サービスサイトは番組音声ファイル生成後に自動で再生成
 - 各 Podcast サービスへの番組配信も RSS 再生成で自動化

こだわりポイント (4/n)

低コストでの運用を目指す

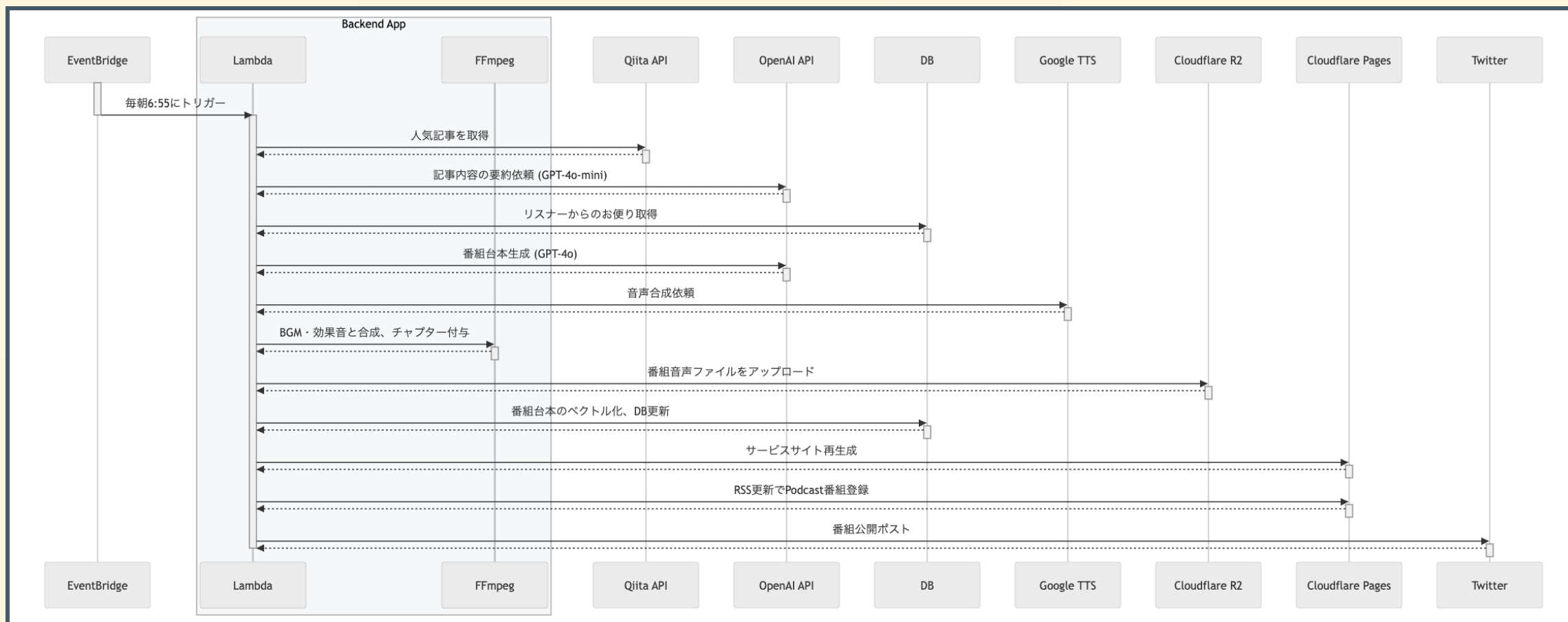
- 人手

- 毎日配信でも人手はかけず完全自動化
- サーバーレス構成で運用の手間も最小限に
- 番組の再生成や用語登録は管理用APIで簡単実行

- 費用

- 生成AIを使うが費用もできるだけ抑える（月額200円以下）

AI ラジオ番組生成フロー



開発スケジュール

開発スケジュール			
2024.11	12	2025.01	02
★ 11/初 検討開始。頭の中で妄想する ★ 11/20 サービス概要や要件を書き出す ★ 11/29 リポジトリ作成	★ 12/17 番組生成ロジックの初版作成	★ 1/12 サイト生成と Podcast サービスへの番組登録を自動化 ★ 1/14 音声合成を OpenAI TTS から Google TTS へ変更	★ 2/1 Tech Post Cast リリース
妄想 → 要件整理	設計、開発		運用、機能追加

運用コスト (2025.2)

※開発での試用を含める

利用サービス	料金 (ドル/月)	備考
AWS (ECR, Lambda, CloudWatch, CDK etc.)	0.70	ECR: 0.65
Google Cloud (Text-to-Speech)	0.00	約13万文字 / 1M文字
Cloudflare (Workers, Pages, R2)	0.00	
OpenAI API (gpt-4o, gpt-4o-mini, text-embedding-3-small)	0.47	287 requests 1.4M tokens
Neon (PostgreSQL)	0.00	
合計	1.17	≒ 181円

Qiita での解説記事



ご清聴ありがとうございました

