

経営革新のための技術戦略： AI + ローコード + クラウドネイティブ

2024/09 | OutSystemsジャパン株式会社

Agenda

-
- | | |
|----|------------------------------------|
| 01 | OutSystems紹介 |
| 02 | 国内OutSystemsユーザ事例 |
| 03 | なぜOutSystemsが活用されているのか？ |
| 04 | OutSystemsによる、AI・ローコード・クラウドネイティブ融合 |
| 05 | 安心してご利用いただけるプラットフォーム |
-

OutSystems紹介

2001年 ポルトガル リスボンで設立

二十数年の歴史と実績：ローコード分野のリーディングカンパニー



OutSystems, Inc.

- ・ ポルトガル：リスボン
- ・ 米国：ボストン



製品開発およびR&D
(ポルトガル)



OutSystemsジャパン株式会社

- ・ 設立：2017年1月
- ・ 代表取締役社長：三重野 智博
- ・ オフィス：東京都港区六本木1-4-5
アークヒルズサウスタワー9F

79⁺

Countries

21

Industries



500⁺

Global Partners

700K⁺

Community
Members

ローコード
革命

LOW CODE
REVOLUTION

OutSystemsジャパン株式会社 代表 三重野 智博

圧倒的スピードで
実現するイノベーション

ローコードプラットフォームで実現する
これからのシステム開発

DevOpsの導入 拡張性・柔軟性 高度なセキュリティ

マイナビ

※ 記載の数値は2024年1月時点のものです

OutSystems紹介：お客様や、我々の考え

ソフトウェア開発は、
非効率な手作業
に頼りすぎではないか？
インフラも、プログラムも、
運用も、。

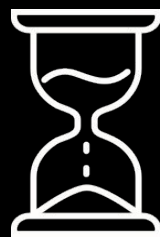


OutSystems紹介：お客様や、我々の考え

ソフトウェアに関連する問題は、
非効率な手作業に頼っている結果
ではないのか？



開発者が足りない



改善するための
時間がとれない



運用ばかり
時間をとられる



作業待ちの項目が
多すぎる

OutSystems紹介：目的

OutSystemsは、
ソフトウェアの
開発・運用作業を
自動化します。

ミッションクリティカル
な領域も、もちろん対
応します。

国内OutSystemsユーザ事例

世界的な大企業様にご活用いただいています。詳細については、ぜひブースまでお越しください。

- [自動車] - トランスフォーメーションの実現
 - <https://www.outsystems.com/ja-jp/case-studies/toyota-low-code-journey/>
- [自動車] - レガシーからの脱却・刷新
 - <https://www.outsystems.com/ja-jp/case-studies/mazda-legacy-migration-to-save-millions/>
- [半導体製造装置] - 基幹システムの刷新
 - <https://www.outsystems.com/ja-jp/case-studies/tokyo-electron/>
- [情報・通信] - B2E 購買システム
 - <https://www.outsystems.com/ja-jp/case-studies/outsystems-for-sbcs-internal-production/>
- [種苗] - 内製化
 - <https://www.outsystems.com/ja-jp/case-studies/sakata-weeds-out-shadow-it/>
- [建設] - DevOpsの実現
 - <https://www.outsystems.com/ja-jp/case-studies/daiwa-house/>
- [製薬] - DX+内製化
 - <https://www.outsystems.com/ja-jp/case-studies/kobayashi-pharmaceutical-dx/>

* OutSystemsのWebサイトで詳細をご覧ください。

なぜOutSystemsが活用されているのか？

基幹システムクラスの 開発・運用自動化による生産性向上

自動化

なぜOutSystemsが活用されているのか？
高い非機能要件を自動的に満たす仕組み
最適な自動化のバランス

自動？生成 AI？

フル



なぜOutSystemsが活用されているのか？ 生成AIによる大規模な自動化は、まだ不十分



以下のソフトウェアは、どのようなアーキテクチャにすべきか、教えてください。

- ユーザは100万人
- 高いセキュリティと可用性
- 内部と外部のユーザで異なる認可・認証
- 外部監査したい
- ...

なぜOutSystemsが活用されているのか？ 生成AIによる大規模な自動化は、まだ不十分

25個の製品を組み合わせ、検証する必要がある。誰がどうやるのか？



アーキテクチャの概要

1. ユーザーアクセスと認証

- **Amazon Cognito**: 外部ユーザーの認証と認可に使用。Amazon Cognitoは、ユーザープールとIDプールを提供し、ソーシャルIDプロバイダーやSAMLベースのIDプロバイダーと統合可能です。
- **AWS IAM**または**AWS SSO** (AWS Single Sign-On): 内部ユーザー（社員など）の認証と認可に使用。IAMは、きめ細かいアクセス制御を提供し、AWSリソースへのアクセスを管理します。

2. フロントエンド (Webとモバイルアプリケーション)

- **Amazon CloudFront**: 高速なコンテンツ配信とキャッシュ。
- **Amazon S3**: 静的コンテンツ（画像、CSS、JavaScriptなど）をホスト。
- **Amazon API Gateway**: APIリクエストのエントリポイント。Webとモバイルアプリケーションの両方が利用可能。

3. バックエンド (ビジネスロジック)

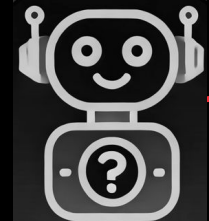
- **AWS Lambda**: サーバーレスでスケーラブルなコンピューティング。必要に応じて自動的にスケールし、メンテナンスが不要です。
- **Amazon ECS** または **EKS**: コンテナベースのアプリケーションをホスト。Dockerコンテナを使用して、マイクロサービスアーキテクチャをサポート。
- **AWS Elastic Beanstalk**: 簡単にWebアプリケーションをデプロイ。特に初期のプロトタイピングやPaaSとして使用する場合に有用。

4. データベースとデータ管理

- **Amazon RDS (Relational Database Service)**: 高可用性とスケーラビリティを備えたリレーショナルデータベース（例えば、MySQL, PostgreSQL, Aurora）。
- **Amazon DynamoDB**: 高速なNoSQLデータベースで、低レイテンシーのデータアクセスを提供。大量のデータをスケーラブルに処理可能。
- **AWS Secrets Manager**: データベース接続情報やAPIキーなどのシークレット情報の管理。
- **AWS Key Management Service (KMS)**: データの暗号化とセキュリティの向上。

5. データの連携

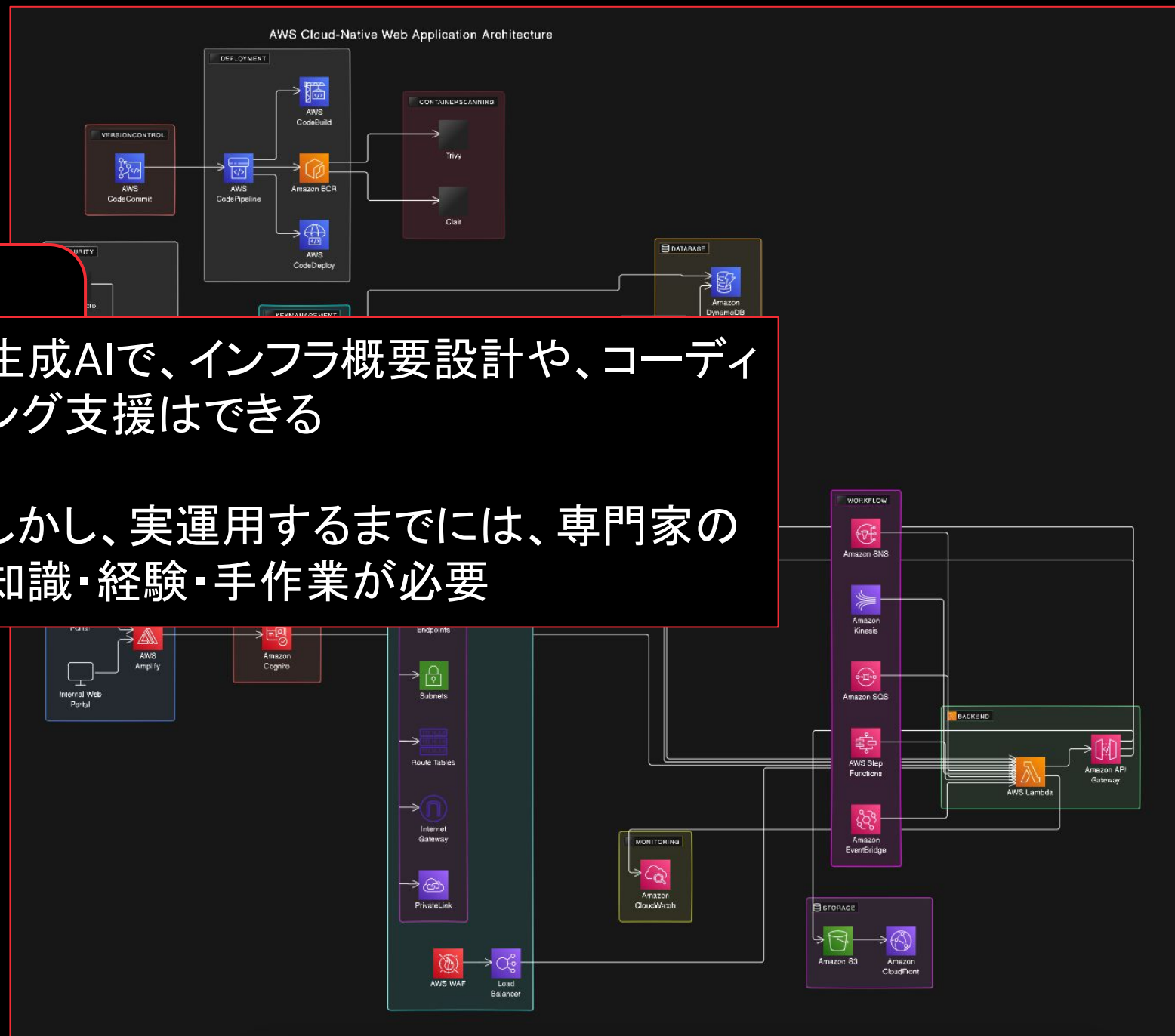
- **AWS Direct Connect**または**AWS VPN**: AWS外部にあるデータベースとの安全で低レイテンシーの接続を確保します。



設計案です

合ってるかどうか
わかりません

- 生成AIで、インフラ概要設計や、コーディング支援はできる
- しかし、実運用するまでには、専門家の知識・経験・手作業が必要



なぜOutSystemsが活用されているのか？ 生成AIによる大規模開発自動化は、まだ不十分



この要求を実現するためには、どのような専門家が必要でしょうか？

なぜOutSystemsが活用されているのか？

生成AIによる大規模開発自動化は、まだ不十分

18～29人の、様々なスキルを持つ人が必要。すぐにトレーニング・採用できるか？

必要なチームメンバーの種類と推奨人数

1. **クラウドアーキテクト**
 - 人数: 1～2人
 - 役割: AWSアーキテクチャ設計、サービス選定、ネットワーク設計、セキュリティ対策。
2. **DevOpsエンジニア**
 - 人数: 2～3人
 - 役割: インフラ自動化、CI/CDパイプライン構築、環境管理、自動スケーリング設定。
3. **バックエンドエンジニア**
 - 人数: 3～4人
 - 役割: バックエンドロジックの開発、API開発、データベース管理。
4. **フロントエンドエンジニア**
 - 人数: 2～3人
 - 役割: Webとモバイルアプリケーションの開発、API統合、UI/UX設計。

1. **セキュリティエンジニア**
 - 人数: 1～2人
 - 役割: セキュリティ設計、IAM管理、データ暗号化、コンプライアンス対応。
2. **データエンジニア**
 - 人数: 1～2人
 - 役割: データベース設計、データパイプライン構築、外部データ連携。
3. **監視・オブザーバビリティエンジニア**
 - 人数: 1人
 - 役割: モニタリング設定、ログ収集と分析、パフォーマンス監視。
4. **プロジェクトマネージャ**
 - 人数: 1人
 - 役割: プロジェクト管理、進捗追跡、リソース管理、チーム間の調整。
5. **インテグレーションエンジニア**
 - 人数: 1～2人
 - 役割: 外部サービスとの統

1. **コンプライアンススペシャリスト**
 - 人数: 1～2人
 - 役割: コンプライアンス遵守の監視、ポリシー設定、監査準備。
2. **テクニカルライター /ドキュメンテーションスペシャリスト**
 - 人数: 1～2人
 - 役割: 技術文書作成、マニュアル・ガイドの作成、ドキュメント管理。
3. **トレーニングスペシャリスト /インストラクター**
 - 人数: 1～2人
 - 役割: トレーニングプログラム設計、教材作成、トレーニング実施。
4. **QAエンジニア(品質保証エンジニア)**
 - 人数: 2～3人
 - 役割: ソフトウェアテスト、自動化テスト、品質保証プロセスの管理。
 - サービス選定、ネットワーク設計、セキュリティ対策。
2. **DevOpsエンジニア**
 - 人数: 2～3人



なぜOutSystemsが活用されているのか？ 重要なシステムに適用できる、実用的な自動化

Outsystem AI

AI をプロセスに組み込み、自動化

OUTSYSTEMS プラットフォーム

Scale | Security | Performance | Flexible Deployment

調査分析

設計

開発

テスト

配置

監視・保守運用

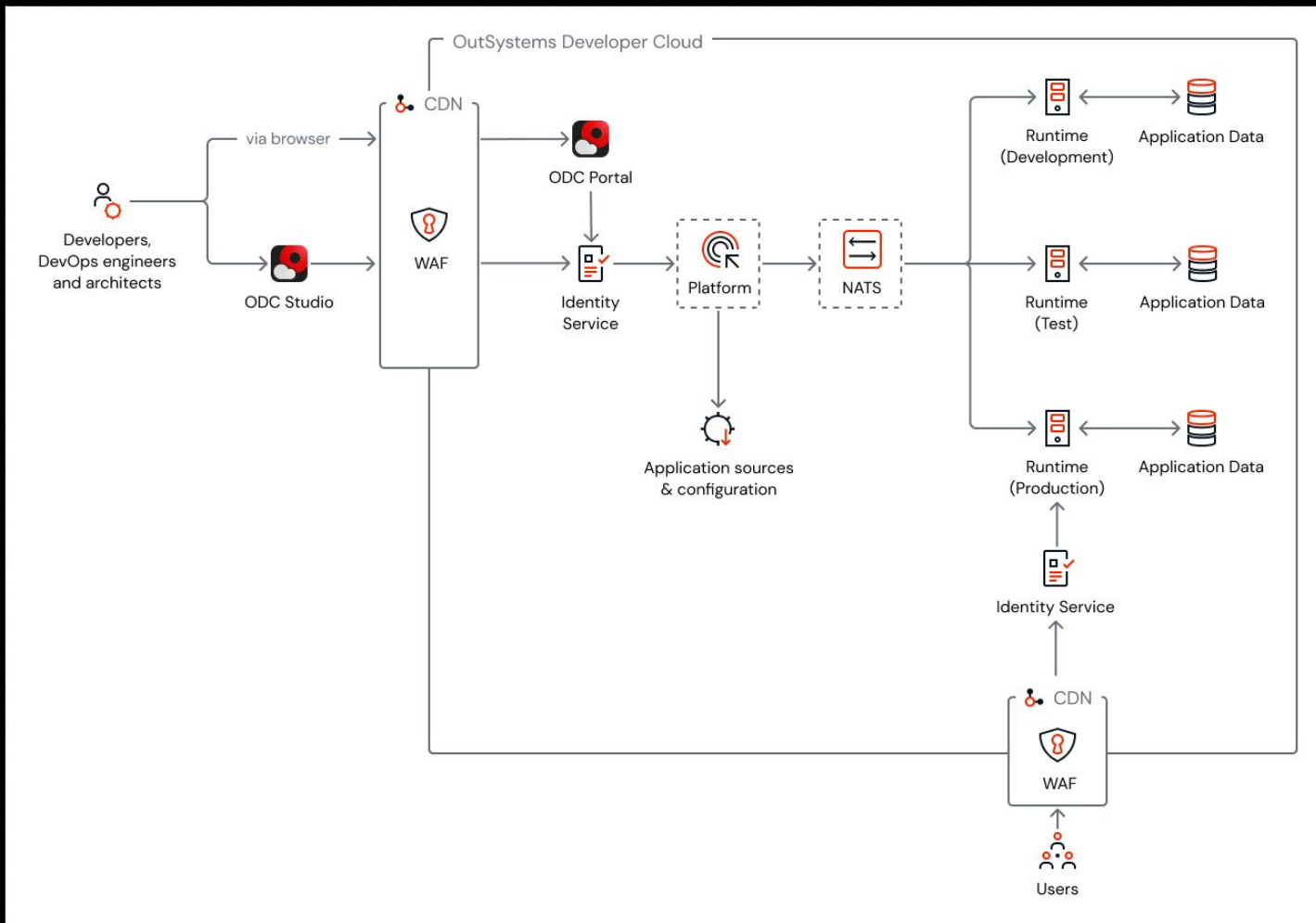
- Product Owner
- Product Manager
- Business Analyst
- CTO
- Enterprise Architect
- Solution Architect
- 1x UX/UI Designer
- 1x Security Architect
- 2 x Front End Engineer
- 2 x Back End Engineer + Data Engineer
- 1 x Dev Manager
- 1 x Dev Team Lead
- QA Automation
- 3 x Tester
- ½ x DevOPS
- Data Administrator
- Support Engineer
- Pilot Users

プラットフォーム組み込みの AI による、GDPR等に準拠した AI ガバナンス



AI・ローコード・クラウドネイティブ融合： 実証済みクラウドネイティブアーキテクチャ

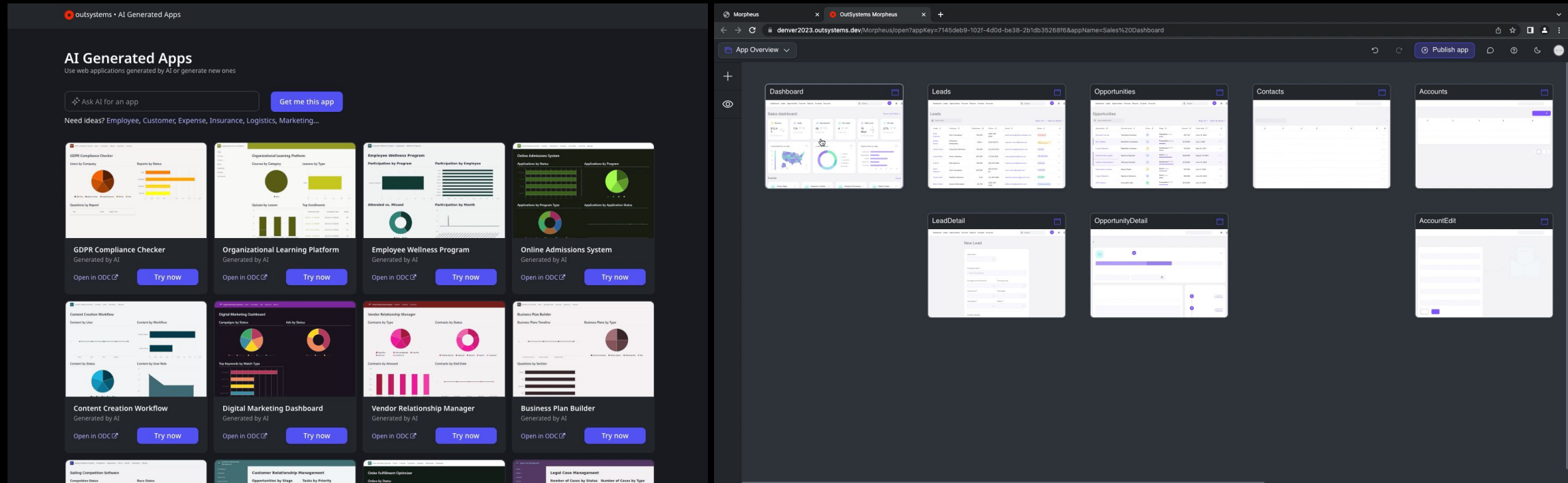
- 各種コンプライアンス対応
- 暗号化、鍵管理
- ユーザ管理
- ネットワークセキュリティ
- 計装
- CI/CD
- プラットフォーム・ランタイム分離
- 自動スケールアウト・自動スケールアップ
- 2億円・12か月以上の工数削減を実現



- 自動セキユアコーディング
- 自動アプリケーション認証・認可
- 自動データ保護
- 自動インフラセキユリティ(※)
- 自動各種セキユリティポリシー準拠と作業(※)

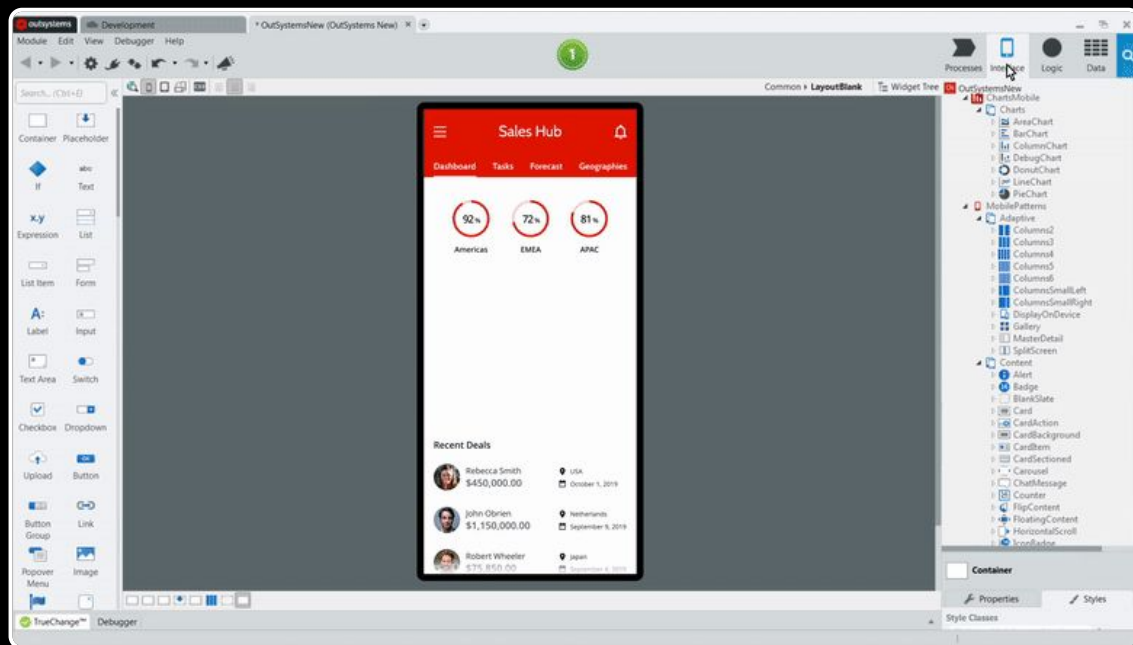


AI・ローコード・クラウドネイティブ融合： プロトタイピング自動化

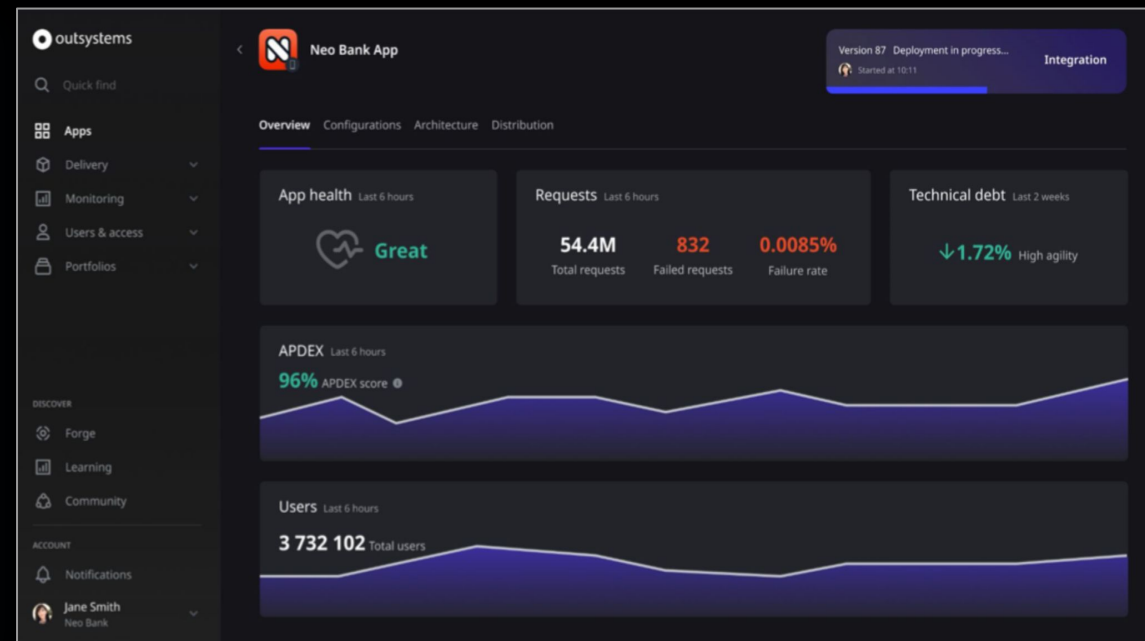


Morpheus :
フルスタック設計・プロトタイピング

AI・ローコード・クラウドネイティブ融合： フルスタック開発・監視・運用

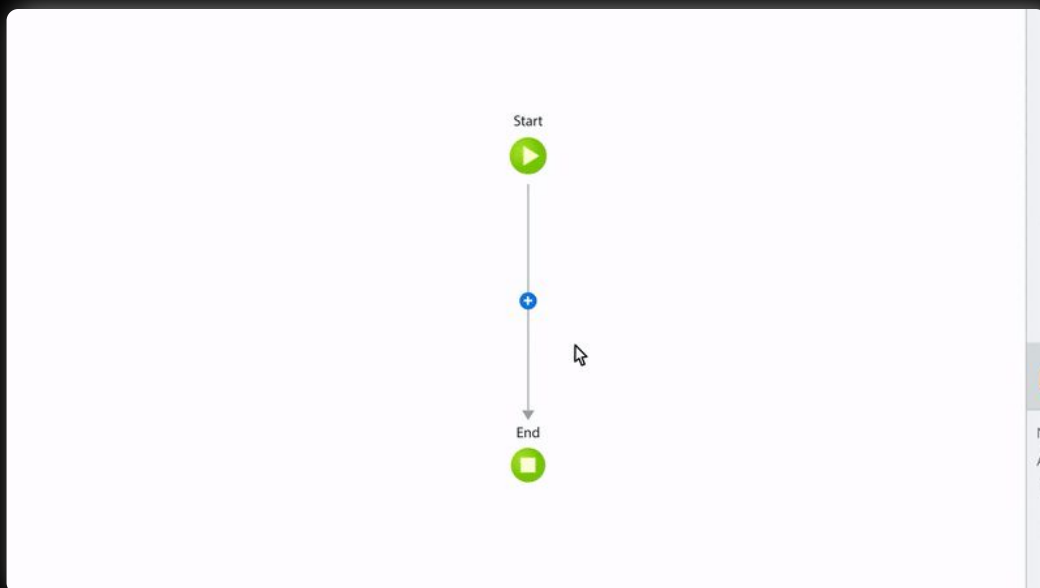


ビジュアル開発・改善



アプリケーションの健康状態監視
ユーザ管理
アプリケーション管理

AI・ローコード・クラウドネイティブ融合： 自動ソフトウェア構造分析・パフォーマンス改善



コードメンター：ロジック生成支援



アーキテクチャメンター：
アーキテクチャ改善支援

AI・ローコード・クラウドネイティブ融合： 自動パフォーマンス・セキュリティ改善

[Back to Apps](#)

Report

Last sync was 2 hours ago. ⓘ

Teams

Apps

Modules

Category

Analyzed

Findings status

User

Clear all

Code patterns	Findings	Technical debt ↓
Unlimited records in Aggregate	14	46.7%
Insecure usage of GetUserId function on client context	2	20.0%
Aggregate or SQL query inside a cycle	2	13.3%
Sequence of connected Aggregates	1	6.7%
Large image	1	6.7%
Missing description on public element	11	3.7%
Duplicated Code	3	3.0%

Aggregate or SQL query inside a cycle

An Aggregate or SQL query is being executed inside a For Each cycle.

Impact

Executing Aggregates and SQL queries inside a cycle can have severe performance impact due to database communication overhead repeated at each iteration. The impact can be greatly worsened when iterating through a list with a high number of elements or when having nested cycles.

How to fix

Avoid executing Aggregates or SQL queries inside a For Each cycle. Instead, replace that Aggregate or SQL query by a more complex one that gets all the related information and execute it before the cycle.

In case you have an Aggregate to fetch the master entity before the cycle followed by another Aggregate inside the cycle to fetch the details, consider eliminating the inner Aggregate by adding a join to the outer Aggregate.

After fixing the finding, set the status to "Already fixed". On the next sync, the finding disappears if you solved it correctly. If not, the finding's status reverts to "Open".

[Learn more about fixing findings](#) ⓘ

Findings	Status
🔗 🔍 📄 GetDepartmentById	Open
🔗 🔍 📄 GetDocumentById	Open

パフォーマンス・セキュリティメンター：
非機能要件の改善支援

AI・ローコード・クラウドネイティブ融合： 生成AI活用作業の自動化

AI Agent Builder My Agents Analytics Configurations ② AH Akira Hirose

< Back to Home

20240607JapanBot Amazon Bedrock - Claude V2 ID:20240607japanbot How to use agents Cancel Save agent

Instruct your agent

Role

OutSystemの技術仕様や用途について詳しい人

Describe the agent area of expertise.

Task

OutSystemsへの質問に、シンプルな日本語で回答してください。回答に利用したデータのURLも教えてください。

Describe how your agent operates and behaves.

Add your data

Data source

LM - RFP Data

Number of records to use

1

Data source items used to generate the answer.

20240607JapanBot Monday 4:35 PM

OutSystemsは、大企業の基幹システム開発にも利用できるプラットフォームです。

OutSystemsの特徴として、以下のような点が大企業の基幹システムに達しています。

- 高いパフォーマンスとスケーラビリティ
OutSystemsは高負荷なシステムに対応できるよう設計されており、垂直・水平スケーリングが可能です。
- 企業向けのセキュリティ機能
ロールベースのアクセス制御、データ暗号化、脆弱性管理など、企業セキュリティ基準を満たす機能を提供しています。
- 既存システムとの連携機能
SOAP、REST APIなどを利用した既存システムとの連携が可能です。
- 信頼性と保守性
24時間365日の運用を支えるサポート体制や災害時のデータ復旧機能などを提供しています。
- アジャイル開発手法の採用
イテレーション開発、継続的デリバリー、自動テストなど、アジャイル開発を支援する機能があります。
- ローコード開発プラットフォーム
開発の効率化と保守性の向上が期待できます。

OutSystemsは、大手金融機関や公共機関などでの導入実績があり、基幹システム開発に利用できると考えられます。

Clear conversation

Ask me anything...

AI-generated answers can have mistakes. [Read more.](#) Token count: 32691

AI Agent Builder My Agents Analytics Configurations ② AH Akira Hirose

Create an agent that...

View all

- Acts as a customer support**
Answer customer issues before they need to open a support ticket.
- Summarizes sales calls**
Summarize sales calls in a JSON format response and fill in MEDPICC steps.
- Acts as an employee support**
Enhance workplace efficiency and automate tasks like writing emails.
- Translates product descriptions**
Translate product descriptions from one language into another.
- Summarizes documentation**
Generate a single-paragraph summary of long documents or articles.
- Generates technical reports**
Generate a complete technical report of a topic from key points.

What's an agent

- Powered by an AI model**
The model is the core of an agent. Start with the available trials or configure your own.
- Infused with your own data**
Bringing in your own data sources is essential to enrich the agent answers.
- Instructed according to your needs**
Define its role, tasks it should perform, tone and how it should react to your inputs.

Agent Builder :
生成AI各種設定・検索拡張生成用データ指定・
テスト・組み込み・サンプルアプリケーション

AI・ローコード・クラウドネイティブ融合： 自動レガシーマイグレーション

詳細分析

コーディング



現行アプリの
コード資源

分析とマッピン
グの定義

自動生成用の
タデータ

GenFactor®

自動生成された
アプリ

開発者による
コーディング

再構築されたア
プリ

安心してご利用いただけるプラットフォーム 開発者の高い評価



批判 (0-6)

8%



中立 (7-8)

30%



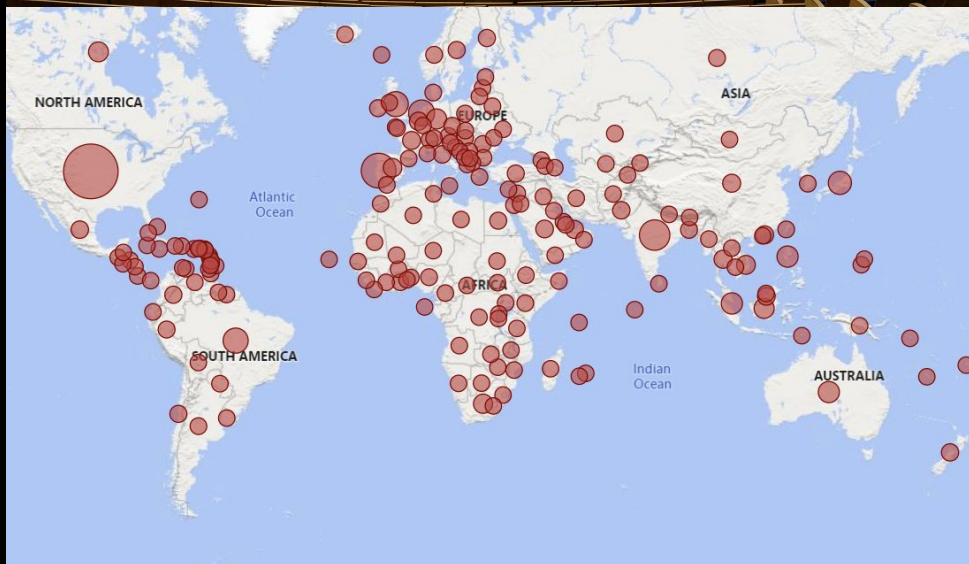
推奨(9-10)

62%

=54
NPS

出典: 2021 OutSystems Community Developer Survey | Collected from a total of 1,126 responses,

安心してご利用いただけるプラットフォーム 70万人以上のユーザコミュニティ




outsystems • User Groups
Find Groups
Upcoming Events
Become an Organizer
Call for Speakers
Community
Log in



Asia Pacific

Nagoya

[Join](#) 106 members

こちらは、OutSystemsに関心のある人のためのグループです。GartnerとForresterがリーダーとして認めたOutSystemsのローコード開発プラットフォームは、全世界で1400社以上に導入されています。Webおよびモバイルのアプリケーションをビジュアルで設計し開発することが可能なローコードプラットフォームでは、一行のソースコードも書く必要がなく、開発者全員が同じレベルのスキルセットで開発することが出来ます。サーバー側の言語、Objective-C、Swift、Javaなどの知識は不要です。また、「ローコード」であるため、コーディングによるカスタマイズも簡単に行うことが出来ます。必要に応じて、従来のコーディングを行うこともできるのがローコードの特徴とも言えます。OutSystemsのローコード開発プラットフォームでは、ビジュアルで開発することによって、従来の開発手法と比較して、5〜6倍の生産性の向上が可能となっています。ぜひGartner Peer Insights、Capterra、TrustRadius、G2 Crowd、GetAppなどのレビューサイトで、満足しているユーザーのレビューをご確認ください。

今年はZoomでのミートアップになります。ご希望の開催時間が有りましたら是非お知らせ下さい。ミートアップでは、チュートリアルやトレーニング、OutSystemsプロジェクトの議論、Q&Aセッションを行います。また、近日中にハッカソンを開催する予定もありますので、ぜひご期待ください。

Hashtag: #OutSystemsUserGroups #OSUGNagoya


outsystems • User Groups
Find Groups
Upcoming Events
Become an Organizer
Call for Speakers
Community
Log in



Asia Pacific

Tokyo

[Join](#) 233 members

ローコード開発ツールであるOutSystemsに興味があるメンバーが集まり、OutSystemsに関する知識を深めるための東京開発者コミュニティです。GartnerとForresterがリーダーとして認めたOutSystemsのローコード開発プラットフォームは、親しみやすいレクランドファイルの開発環境、ビルド済みテンプレート、再利用可能なコンポーネント、AIによるガイダンス、カスタムコードによる拡張、そして多彩な最新テクノロジー（ML、IoT、RPA、マイクロサービスなど）すべてを備え、全世界で数千社に導入されています。Webおよびモバイルのアプリケーションをビジュアルで設計し開発することが可能なローコードプラットフォームでは、ソースコードを書く必要がなく、開発者全員が同じレベルのスキルセットで開発することが出来ます。サーバー側の言語、Objective-C、Swift、Javaなどの知識は不要ですが、必要に応じて、コーディングによるカスタマイズも簡単に行うことが出来ます。OutSystemsのローコード開発プラットフォームでは、従来の開発手法と比較して、5〜6倍の生産性の向上が可能となっています。ぜひGartner Peer Insights、Capterra、TrustRadius、G2 Crowd、GetAppなどのレビューサイトで、満足しているユーザーのレビューをご確認ください。

ミートアップでは、チュートリアルやトレーニング、OutSystemsプロジェクトの議論、Q&Aセッションを行います。主に東京での開催を予定しています。他県からの参加も大歓迎です！

ありがとうございました。
是非ブースでお話ししましょう!
