



SPI Japan 2023 セッション2A

サービス開発初心者チームの運用を意識したDevOps開発について

株式会社インテック

クラウドプロダクトサービス部

木村慎吾

■ 概要

- ・ サービス開発・運用の初心チームの改善内容の事例紹介

■ お伝えしたいこと

- ・ サービス開発時に意識したことが運用開始後の改善にも役に立った
- ・ リリース後も開発と運用を分離させない工夫について



木村 慎吾

インテック
クラウドプロダクトサービス部

開発との関わり

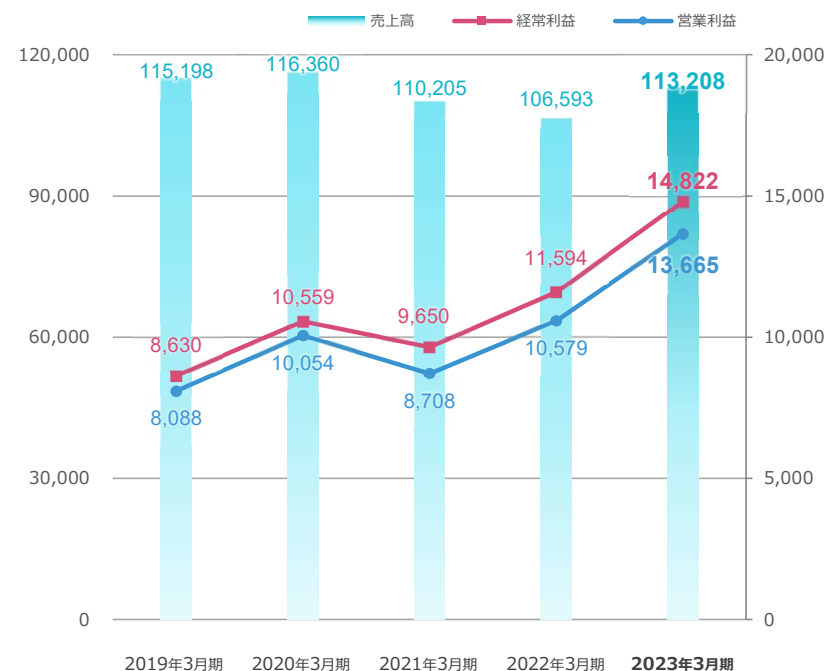
- ・ サービスオーナーとして従事
- ・ ここ数年はDevOpsを推進

インテックは、1964年の創業から半世紀に渡り、「いつでも、どこでも、誰もが」自由にコンピュータの恩恵を受けられるコンピュータ・ユーティリティ社会の実現を目指し、多くのチャレンジを続け、お客さまビジネスと社会の発展に貢献しています。

社名	株式会社インテック (INTEC Inc.)
設立	1964年1月11日
資本金	208億30百万円
代表者	代表取締役社長 北岡 隆之
従業員	3,897名 (2023年6月1日現在)
株主	TIS株式会社 (100%)
主要取引銀行	三菱UFJ銀行 富山第一銀行 みずほ銀行 北國銀行

■ 業績推移

(単位：百万円)



商号の由来



INTECの社名はつぎのような意味を込めてつくられました。

INFORMATION TECHNOLOGY (情報技術)

増大するデータから価値ある情報を創造し提供する技術

INTERNATIONAL TELE-COMMUNICATION (国際情報通信)

ナショナルからインターナショナルへ、グローバルな情報通信技術

INTEGRATED TECHNOLOGY (システム技術)

未来を予見する情報技術をベースとし、各種各様な科学・工学を包括する技術などの略称であり、またそうしたすべての分野に意欲的に取り組む集団として

INTELLECTUAL ECHELON (創造的知的集団)

を象徴するものであります。

創業の精神

私の信条
奉仕こそわが務め
創造こそわが喜び
世界こそわが職場
敏業こそわが望み
信心こそわが行状
社長

1971年 創業者金岡幸二が社長訓示として披露

事例について

■ サービス概要

- ・ サービス名：ID認証サービス
- ・ リリース日：2021年4月（開発期間：2019年11月～2021年2月）



報道関係各位
2021年04月07日
株式会社インテック

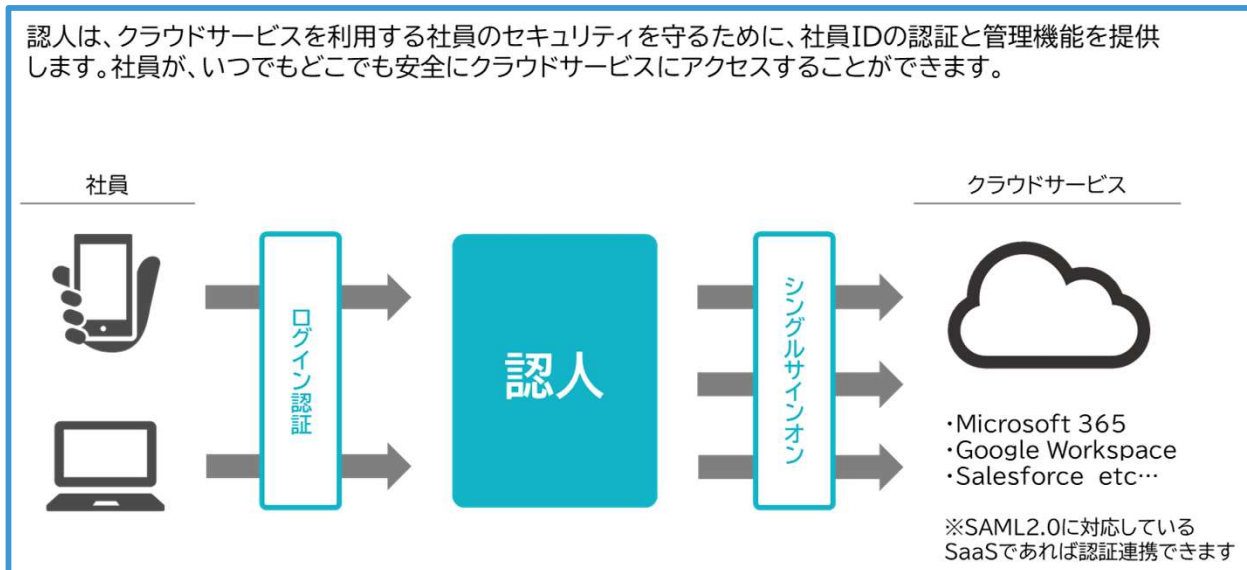
インテック、複数クラウドサービスのID認証を一括で管理できる「ID認証サービス」に新機能を追加～多要素認証やユーザ管理機能を提供～

TISインテックグループの株式会社インテック（本社：富山県富山市、代表取締役社長：北岡隆之、以下インテック）は、複数クラウドサービスのID認証を一括で管理できるクラウド型「ID認証サービス」に、よりセキュアな認証を実現する多要素認証機能、クラウドサービス連用をスムーズに行うためのユーザ管理機能やモニタリング・レポート機能などを追加し、本日より提供することを発表します。

「ID認証サービス」は、複数クラウドサービスのID認証を一括で管理できるクラウド型の認証基盤サービスで、クラウドサービスを開発するサービス事業者や、クラウドサービスの利用を推進する企業の情報システム部門の方などにご利用いただいております。

今回新たに追加した、証明書認証やワンタイムパスワード認証などの多要素認証機能を利用することで、よりセキュアな認証が可能となります。

また、クラウドサービス開発者は、ユーザ管理機能のAPIを利用することで、認証機能の開発ノウハウがなくとも自社サービスでセキュリティレベルの高い認証機能が提供可能となります。自社で認証機能の開発の必要がないため、自社サービスのコア開発に専念することができます。



■ 体制

- ・ サービスオーナー：木村
- ・ 開発メンバ：9名
- ・ 運用メンバ：2名
- ・ 営業・企画など：数名

■ チームの特徴

- ・ 過去のパッケージ開発でアジャイル開発の経験はあり
- ・ サービスの開発・運用は今回が初めて
⇒リリースから2年経過

運用後の改善活動について

リリース後もDevOpsを継続するために主に3つの施策を実施

施策	タイミング	概要	参加者
朝会	毎朝	タスクの確認やシステムのモニタリングを実施する	オーナー、開発メンバ、運用メンバ
ふりかえり	2週間ごと	2週間で発生した課題などを整理し改善タスクなど作成する ⇒本日ここから出た改善内容について紹介	開発メンバ、運用メンバ
サービスマネージメントふりかえり	4半期ごと	サービスマネージメント目線で課題などを整理し改善施策を作成する	オーナー、開発リーダ、運用リーダ

朝会では以下のような画面を見ながら確認

■ モニター



■ カンバン



■ ふりかえり



DevOpsへの意識について

■ DevOpsの取り組みへの経緯

サービス開発の開始にあたり課題として以下のような問題があった

■ ■ 問題点

- ・ これまでのやり方は開発工程後に運用工程へ引き継がれ分断されていた
 - ・ バラバラに設計され、それぞれの観点で対応していた
 - ⇒開発と運用でギャップが発生し、スピード感がなくなる
 - ⇒部分カスタマイズが増え、長期運用になると構成が複雑になる

■ ■ やりたいこと

- ・ スピード感を持った対応
- ・ シンプルな構成の維持し継続的な開発を実施

⇒やりたいことを実現する手段としてDevOpsを採用した

■ DevOps採用時の課題

- スタート時には見本が無かったため、どこから初めてよいかわからなかった
- ・ DevOpsにはスクラムのようなフレームワーク的ものが無かった
 - ・ 自社の事例も無かった

■ ポイント

模索しながら開発を進める中で以下の2点が進める糸口になった

- 1 : マネジメントシステムを意識した取り組み
- 2 : システムテスト工程の設置 (※継続を意識した取り組み)

⇒このポイントはリリース後の改善においても重要な指針になってくれた

1 : マネジメントシステムを意識した取り組み

1 : マネジメントシステムを意識した取り組みについて

マネジメントシステムの中でも、開発と運用の関連が強いプロセスが重要である。

ポイントとしては以下の2点がある。

- ① : リリースに向けた一連のフローについて
- ② : ログ・監視モニタリングについて

1点目の「リリースに向けたフローについて」

■スタート時から意識していたこと

- ・インシデント管理、構成管理、リリース管理の流れを意識して管理・運用

■運用開始後の課題について

- ・リリース後に発生した障害の調査に時間がかかりすぎている
 - ・各工程の関連がすぐに把握できていない
 - ・リリースしたアウトプットの影響範囲がすぐに把握できない

① : リリースに向けた一連のフローについて 2/3

■ 取り組みについて

- ・ 開発時に設計したフローや管理の概要としては以下のことを実施していた

(課題確認→修正→リリース)

インシデント発生 → リリース

(修正結果を反映)

構成管理

リリースの展開結果のみ

20210512リリース(Bugfix:Webapp)

リリース結果

- 成功

Bugfix

- yuito-sakuto/ninto-webapp

Pipeline

- https://ap-northeast-1.amazonaws.com/codebuild/build-logs/pipeline/pipelines/ninto-webapp-pipeline/executions/...

Assets

ビルドステータス

ビルドログ

```
1 [Container] 2023/07/25 09:53:08 waiting for agent doing
2 [Container] 2023/07/25 09:53:10 waiting for DOWNLOAD_SOURCE
3 [Container] 2023/07/25 09:53:13 Phase is DOWNLOAD_SOURCE
4 [Container] 2023/07/25 09:53:17
   CODEBUILD_SRC_DIR=/codebuild/output/arc08444977/src/github.com/yuito-sakuto/ninto-webapp
5 [Container] 2023/07/25 09:53:17 Yarn Install
   /codebuild/output/arc08444977/src/github.com/yuito-sakuto/ninto-webapp/builddspec.yml
```


①：リリースに向けた一連のフローについて 3/3

■改善策について

- ・リリース時に構成管理や状況などインシデントなどに関連する内容を詳細に記載する

■ポイント

- ・開発と運用で関連するプロセスは開発時から意識しておく

⇒開発時に各プロセスごとのフローがあったので、改善はスムーズにできた



2点目の「ログ・監視モニタリングについて」

■ スタート時から意識していたこと

- ・ 安易な運用回避対応を避けるため、開発・運用チーム一緒にモニター画面などで運用状況を確認
- ※特に無駄なログが出ていないことを意識して運用をスタート

■ 運用開始後の課題について

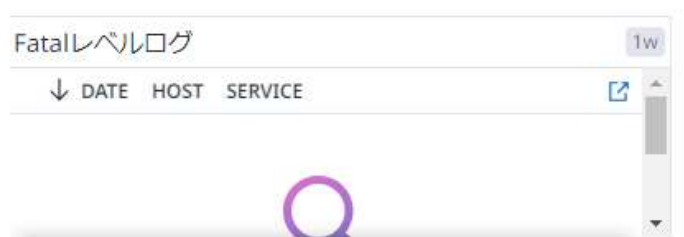
- ・ 処理が動いていない場合の障害検知ができていなかった
 - ・ エラーログ検知が中心だった
 - ・ 動作していない場合に対する考慮不足

② : ログ・監視モニタリングについて 2/3

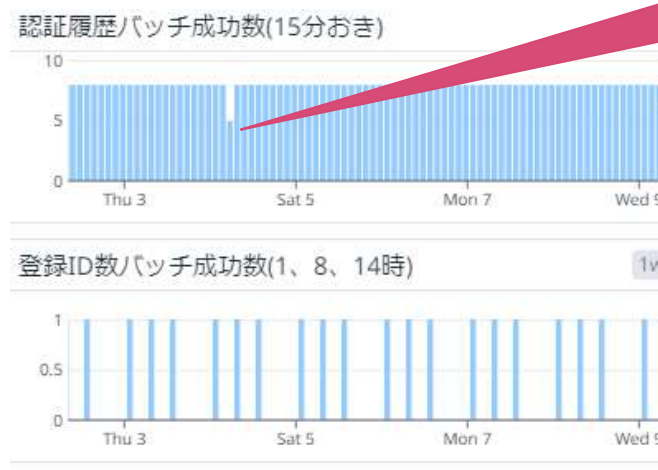
■ 取り組みについて

- ・ FATALログなどの障害発生時のみに注目してモニタリングを開始
⇒ 正常処理数を数えて、処理されていない場合への対応を追加

・ 障害ログ監視画面



・ 処理実行監視画面



動いていないことを監視

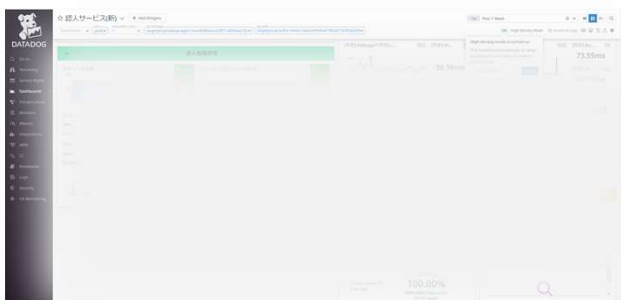
② : ログ・監視モニタリングについて 3/3

■ 改善策について

- ・ 正常時の数に処理数達していない状態を検知する設定をモニター画面に追加

■ ポイント

- ・ モニター画面などの確認を開発側と運用側が一緒に行える場をつくる



日々、負荷状況や
リソース消費量も
監視

⇒開発・運用チームのどちらか一方のみで勝手に判断した対応をしない

2 : システムテスト工程の配置

- スタート時から意識していたこと
 - ・ 機能ごとにリリースできるようにしたかった
 - ⇒しかし、サービス開発時はウォーターフォール的に最後にビッグバンリリースとしてシステムテスト工程を設けた

- 運用開始後の課題について
 - ・ サービスリリース後もビッグバンリリースのままではスピード感が出ない
 - ⇒追加開発においては、機能をまとめてリリースすることはサービスの成長にとってマイナス要因であったため、同じやり方で進めることはできないと考えた

■ 改善策について

- ・ 機能を1つ1つリリースできるように、負荷テストなど用途別にテスト環境を提供できるようにした

■ ポイント

- ・ 開発時から理想形を意識した対応をしていた
 - ・ クラウドネイティブ開発によって、インフラもコード化や自動化をしていた
 - ・ 開発期間を通してチームが成長し、ノウハウが蓄積された
 - ・ リリース方針もこのタイミングで見直し、各種リリースの基準を小さくリリースできるように再作成した

⇒ チームとして常に継続してサービスの開発・改善に取り組んだDevOpsの成果だと考えている

リリースごとの方針

サービス: アプリケーション(Auth/Webapp)、インフラ

リリースの種類	デプロイタイプ (ユーザー影響)	変更分類	リリース承認に必要な条件	リリース展開承認者	例
エンハンスメント「重大」	重大	通知	展開計画 ↳ ユーザーへのアナウンス実施(対象: 主要ユーザー)	サービス担当者	サービス停止やデータ移行を伴う場合は機能追加・変更
エンハンスメント「通常」	大	通知			ユーザー利用に影響を与える機能追加・変更 ↳ ユーザー利用面の機能追加・変更
エンハンスメント「通常」	小	通知			ユーザー利用に影響を与えない機能追加・変更 ↳ エンドシステム側の機能追加・変更
障害対応「緊急」	重大	緊急	緊急対応計画(対象: 主要ユーザー) ↳ 実施承認		サービスの中核を回復するためのリリース

まとめ

■本日のポイント

サービス開発段階から以下を意識することで、運用後の改善もスムーズになる

- 1：マネジメントシステムを意識した取り組み
- 2：システムテスト工程の設置（※継続を意識した取り組み）

■現状について

サービスリリース後、追加機能開発もしながらサービス運営を約2年継続

- ・この直近半期のインシデントは0件で安定稼働している
- ・最初から100%うまくいっていたわけではなく改善しながら成長中
 - ・最初の半期で顧客影響はないがインシデントが約30件発生
 - ・最近の半期でヒヤリハットレベルの課題は約25件発生

■全体総括

- ・ 初期開発時から運用を意識させることは重要
 - ・ 開発チームの運用に対する意識改革
 - ・ 継続的な活動への意識改革
- ・ DevOpsとして開発と運用を分断させない工夫も必要
 - ・ 朝会やふりかえりなどを一緒にやることで一体感を出す
 - ・ サービス全体を俯瞰して、組織としてDevOpsが機能しているかも確認

ご清聴ありがとうございました

ITで、社会の願い叶えよう。



<本資料に関するお問い合わせ >

株式会社インテック
ネットワーク&アウトソーシング事業本部 クラウドプロダクトサービス部
E-Mail : kimura_shingo@intec.co.jp

<本資料の取り扱いに関して>

本資料は、著作権法及び不正競争防止法上の保護を受けております。資料の一部あるいは全部について、株式会社インテックから許諾を得ずに、複写、複製、転記、転載、改変、ノウハウの使用、営業秘密の開示等を行うことは禁じられております。本文記載の社名・製品名・ロゴは各社の商標または登録商標です。