



Red Hat Mobile Application Platform

ホスト型 3 製品の機能

Red Hat Mobile Application Platform ホスト型 3 向け

Red Hat Customer Content
Services

Red Hat Mobile Application Platform ホスト型 3 向け

法律上の通知

Copyright © 2017 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

本ガイドでは、Red Hat Mobile Application Platform ホスト型 3 の機能に関する包括的な説明と使用方法を提供しています。

目次

第1章 プロジェクト、アプリ、およびサービス	8
1.1. プロジェクト	8
1.1.1. 概要	8
1.1.2. 新規プロジェクト	8
1.1.2.1. クライアントのプラットフォーム	8
1.1.3. インポート	8
1.1.4. アプリ、クラウドアプリ & サービス	9
1.1.4.1. アプリ	9
1.1.4.2. クラウドコードアプリ	9
1.1.4.3. サービス	9
1.1.5. アプリをプロジェクトに追加する	9
1.1.5.1. 新規アプリの作成	9
1.1.5.2. 既存アプリのインポート	9
1.1.5.3. 既存アプリのクローン	10
1.1.6. 接続	10
1.1.6.1. SDK Config	10
1.1.6.2. 接続の再設定	10
1.1.6.3. 接続を無効にする	10
1.1.7. Git クイックスタート	10
1.1.8. リソース	11
1.1.9. ライフサイクル管理	11
1.1.10. フォーム	11
1.1.11. レポート	12
1.1.12. 設定	12
1.1.12.1. プロジェクトの削除	12
1.1.12.2. プロジェクトの名前変更	12
1.2. アプリ	12
1.2.1. 概要	12
1.2.2. 詳細	13
1.2.3. エディター	14
1.2.4. Docs	14
1.2.5. アナリティクス	14
1.2.6. プッシュ	15
1.2.7. ビルド	15
1.2.7.1. アプリバイナリーをサードパーティー MAM および MDM プロバイダーに公開する	16
1.2.7.1.1. Apperian	16
1.2.8. エクスポート	16
1.2.9. 認証情報	17
1.2.9.1. 認証情報バンドル	17
1.2.9.1.1. 新規バンドルの作成	18
1.2.9.1.2. 既存の開発リソースの移行	18
1.2.10. 統合	18
1.3. クラウドアプリ	19
1.3.1. 概要	19
1.3.2. 詳細	19
1.3.3. エディター	20
1.3.4. 環境変数	20
1.3.4.1. 環境変数の一覧	20
1.3.4.1.1. アプリの環境変数	20
1.3.4.1.2. デプロイ済みユーザーの環境変数	20
1.3.4.1.3. デプロイ済みのシステム環境変数	21

1.3.4.2. 環境変数の作成	21
1.3.4.3. 環境変数の設定解除/削除	21
1.3.4.4. 環境変数をプッシュ	21
1.3.4.5. 環境変数のコードでの参照	21
1.3.5. データブラウザー	22
1.3.6. デプロイ	22
1.3.7. 統計	23
1.3.7.1. アプリのエンドポイント	23
1.3.7.2. カスタム統計値	23
1.3.7.2.1. カウンター	24
1.3.7.2.2. タイマー	24
1.3.8. 通知およびイベント	24
1.3.8.1. システムイベントビュー	26
1.3.8.2. 警告 & 通知メール	27
1.3.8.2.1. 警告ビュー	27
1.3.8.2.2. 警告の作成	27
1.3.8.2.3. 通知ビュー	27
1.3.9. ログ	27
1.3.10. エンドポイントセキュリティー	27
1.3.10.1. エンドポイントセキュリティーの管理	27
1.3.10.1.1. アプリのセキュリティー	27
1.3.10.1.2. 個別のエンドポイントセキュリティー	27
1.3.10.2. 監査ログ	28
1.3.10.3. CORS サポート	28
1.4. MBAAS サービス	28
1.4.1. 概要	28
1.4.2. MBaaS サービスのプロビジョニング	29
1.4.3. 詳細	29
1.4.3.1. サービスの詳細	29
1.4.3.2. サービス設定	29
1.4.4. Docs	29
1.4.5. エディター	30
1.4.6. 環境変数	30
1.4.6.1. アプリの環境変数	30
1.4.6.2. デプロイ済みユーザーの環境変数	30
1.4.6.3. デプロイ済みのシステム環境変数	30
1.4.6.4. 環境変数の作成	30
1.4.6.5. 環境変数の設定解除/削除	31
1.4.6.6. 環境変数をプッシュ	31
1.4.7. データブラウザー	31
1.4.7.1. データブラウザーを有効にする	31
1.4.7.2. データブラウザーの使用	32
1.4.7.2.1. コレクションの表示/追加	32
1.4.7.2.2. コレクション内のデータ表示	32
1.4.7.2.2.1. データの並び替え	32
1.4.7.2.2.2. データのフィルタリング	32
1.4.7.2.3. データの編集	33
1.4.7.2.3.1. インラインエディターを使用した編集	33
1.4.7.2.3.2. 高度なエディターを使用した編集	33
1.4.7.2.3.3. 動的エディターを使用した編集	33
1.4.7.2.3.4. Raw JSON エディターを使用した編集	33
1.4.8. デプロイ	34
1.4.9. 統計	34

1.4.9.1. アプリのエンドポイント	34
1.4.9.2. カスタム統計値	34
1.4.9.2.1. カウンター	34
1.4.9.2.2. タイマー	35
1.4.10. 通知およびイベント	35
1.4.10.1. システムイベントビュー	37
1.4.10.2. 警告 & 通知メール	37
1.4.10.2.1. 警告ビュー	38
1.4.10.2.2. 警告の作成	38
1.4.10.2.3. 通知ビュー	38
1.4.11. ログ	38
1.4.12. エンドポイントセキュリティ	38
1.4.12.1. エンドポイントセキュリティの管理	38
1.4.12.1.1. アプリのセキュリティ	38
1.4.12.1.2. 個別のエンドポイントセキュリティ	38
1.4.12.2. 監査ログ	39
1.4.12.3. CORS サポート	39
第2章 モバイルバックエンド	40
2.1. MBAAS ターゲットおよび環境	40
2.1.1. 概要	40
2.1.2. MBaaS ターゲットの作成	40
2.1.2.1. FeedHenry MBaaS ターゲット	40
2.1.2.2. OpenShift 2 MBaaS ターゲット	40
2.1.2.3. OpenShift 3 MBaaS ターゲット	41
2.1.3. 環境の作成	41
2.2. クラウドリソース	42
2.2.1. 概要	42
2.2.2. サマリービュー	42
2.2.3. リソース	42
2.2.4. アプリ	42
2.2.5. キャッシュ	43
2.3. プッシュ通知	43
2.3.1. 概要	43
2.3.2. サポートされるプラットフォーム	43
2.3.3. 作動方法	43
2.3.4. 通知の送信	45
2.3.4.1. クライアントアプリのフィルタリング	45
2.3.4.2. 受信者のフィルタリング	45
2.3.5. セキュリティ面の留意点	46
2.3.5.1. 通知のコンテンツ	46
2.3.5.2. バリアントシークレット	46
2.3.6. プッシュネットワーク用認証情報の取得	46
2.3.6.1. Android	47
2.3.6.2. iOS	47
2.3.6.3. Windows	47
2.3.7. アナリティクス	48
2.3.8. プッシュ通知対応の Cordova アプリの構築	48
2.3.9. コミュニティーバージョンの UnifiedPush Server	48
第3章 DRAG AND DROP APPS	49
3.1. フォームビルダーを使用したフォームの作成	49
3.1.1. フォームの作成	49
3.1.2. フィールドをフォームに追加する	49

3.1.2.1. 構造フィールド	50
3.1.2.1.1. Section Break	50
3.1.2.1.2. Page Break	50
3.1.2.2. コンテンツフィールド	50
3.1.2.2.1. テキストフィールド	50
3.1.2.2.2. 段落フィールド	51
3.1.2.2.3. 数字フィールド	51
3.1.2.2.4. E メールフィールド	51
3.1.2.2.5. ウェブサイトフィールド	51
3.1.2.2.6. ドロップダウンメニュー	51
3.1.2.2.7. ラジオボタン	51
3.1.2.2.8. チェックボックス	51
3.1.2.2.9. マップ	52
3.1.2.2.10. 場所	52
3.1.2.2.11. ファイル	52
3.1.2.2.12. フォトキャプチャー	52
3.1.2.2.13. 署名キャプチャー	52
3.1.2.2.14. Datestamp	52
3.2. フォームのプロジェクトの管理	53
3.2.1. 概要	53
3.2.2. フォームプロジェクトの作成	53
3.2.2.1. 新規プロジェクトの作成	53
3.2.2.2. 既存のプロジェクトに関連付ける	53
3.2.3. プロジェクトにフォームとテーマを追加する	54
3.2.4. フォームプロジェクトの設定	54
3.2.4.1. クライアントオプション	54
3.2.4.1.1. カメラの設定	54
3.2.4.2. 提出の設定	54
3.2.4.3. クライアントロギング	55
3.2.4.4. 管理者ユーザー	55
3.2.4.5. クラウドロギング	55
3.3. フォームフィールドおよびページを動的に表示する	55
3.3.1. 概要	55
3.3.2. ルールの定義	55
3.4. データソースの使用	56
3.4.1. 概要	56
3.4.2. 作動方法	57
3.4.2.1. リフレッシュ期間	57
3.4.3. 要件	57
3.4.3.1. JSON 形式	57
3.4.3.2. MBaaS サービスエンドポイント	58
3.5. テーマ	59
3.5.1. テーマの作成	59
3.5.2. テーマの編集	59
3.5.2.1. ロゴ	59
3.5.2.2. 背景	59
3.5.2.3. 文字の体裁	59
3.5.2.4. 境界線	59
3.5.2.5. スペース	59
3.6. 提出の表示	60
3.6.1. 概要	60
3.6.2. 提出の並び替え	60
3.6.2.1. フォームごとの並び替え	60

3.6.2.2. プロジェクトごとの並び替え	61
3.6.2.3. 提出のフィルタリング	61
3.6.3. 提出の検索	61
3.6.3.1. 検索基準の作成	61
3.6.4. 提出詳細の表示	62
3.6.5. 提出の編集	62
3.6.6. 全提出で通知を受ける	63
第4章 運営管理	64
4.1. アカウント	64
4.1.1. アカウントの管理	64
4.1.2. SSH キーの管理	64
4.1.3. API キーの管理	65
4.2. 管理	65
4.2.1. ユーザー	66
4.2.1.1. ユーザーの表示	66
4.2.1.2. ユーザーの作成	66
4.2.1.3. ユーザーの更新	66
4.2.1.4. 招待メール	67
4.2.2. ユーザーロール	67
4.2.2.1. アナリティクス (analytics)	68
4.2.2.2. フォームエディター (formseditor)	68
4.2.2.3. フォーム管理者 (formsadmin)	68
4.2.2.4. 提出ビューワ (submissionsviewer)	68
4.2.2.5. 提出エディター (submissionseditor)	68
4.2.2.6. 開発者 (dev)	68
4.2.2.7. 開発管理者 (devadmin)	69
4.2.2.8. サービス管理者 (serviceadmin)	69
4.2.2.9. ドメイン管理者 (portaladmin)	69
4.2.2.10. 顧客管理者 (customeradmin)	69
4.2.2.11. リセラー管理者 (reselleradmin)	69
4.2.2.12. 管理者 (admin)	69
4.2.3. 認証ポリシー	70
4.2.3.1. 認証ポリシーの表示	70
4.2.3.2. 認証ポリシーの作成	70
4.2.3.3. OAuth 認証タイプ	70
4.2.3.4. LDAP 認証タイプ	70
4.2.3.4.1. URL	71
4.2.3.4.2. DN 接頭辞	71
4.2.3.4.3. DN	71
4.2.3.4.4. DN および DN 接頭辞	71
4.2.3.5. Active Directory の設定例	71
4.2.3.6. FeedHenry 認証タイプ	72
4.2.3.7. MBAAS 認証タイプ	72
4.2.3.7.1. サービス	72
4.2.3.7.2. エンドポイント	72
4.2.3.7.3. デフォルト環境	72
4.2.3.8. 承認	72
4.2.3.8.1. ユーザー終了の確認	72
4.2.3.8.2. ユーザーが承認済みか確認	72
4.2.4. モバイルアプリの管理	73
4.2.4.1. App Store	73
4.2.4.1.1. App Store の設定	73

4.2.4.1.2. App Store の使用	73
4.2.4.2. App Store のアプリ	77
4.2.4.2.1. App Store のアプリの追加	77
4.2.4.2.2. App Store のアプリ更新	77
4.2.4.2.3. バイナリーのアップロード	78
4.2.4.2.3.1. バイナリーのアップロード	78
4.2.4.2.3.2. iOS バイナリー	78
4.2.4.2.3.3. Windows Phone 8 バイナリー	78
4.2.4.2.3.4. バイナリーの履歴	78
4.2.4.3. サードパーティー MAM/MDM	79
4.2.5. 監査ログ	79
4.2.5.1. 検索とフィルターの機能	79
4.2.6. グループ	79
4.2.6.1. 概要	79
4.2.6.2. 新規グループの追加	80
4.2.6.3. グループの編集	80
4.2.7. デバイス	80
4.2.7.1. デバイスの表示	80
4.2.7.2. デバイスの更新	81
4.2.7.3. デバイスユーザーの表示	81
4.2.7.4. デバイス上のアプリの表示	81
4.3. アナリティクス	81
4.3.1. 集計レポート	82
4.3.1.1. 月間アクティブデバイス	82
4.3.1.1.1. デバイス識別子	82
4.3.2. プロジェクト別レポート	82
4.3.2.1. デバイスインストール	82
4.3.2.2. アプリ起動	83
4.3.2.3. クラウドリクエスト	83
4.3.2.4. アクティブデバイス	83
4.3.3. レポート図表 & グラフ	83
4.3.3.1. 集計レポート	83
4.3.3.2. アプリ別レポート	84
4.3.3.3. 個別アプリの分析	84
4.3.3.4. 日付別の分析	84
4.3.3.5. プラットフォーム別の分析	84
4.3.3.6. 場所別の分析	84
4.4. チームおよびコラボレーション	84
4.4.1. 概要	84
4.4.1.1. チームリスト	85
4.4.2. チームの作成	85
4.4.3. ダッシュボード	85
4.4.4. メンバー	85
4.4.5. パーミッション	86
4.4.5.1. 主要な用語	86
4.4.5.2. ビジネスオブジェクト	86
4.4.5.3. アクセスレベル	86
4.4.5.4. 継承	86
4.4.5.5. 複数チーム	87
4.4.5.6. フィルタリング可能なビジネスオブジェクト	87

第1章 プロジェクト、アプリ、およびサービス

1.1. プロジェクト

1.1.1. 概要

プロジェクトセクションでは、プロジェクトの作成と管理ができます。プロジェクトを作成すると、複数のアプリをこれに関連付けることができます。既存のアプリがある場合は、「インポート」機能を使ってこれをプラットフォームにインポートすることができます。

要件

ユーザーは、以下のパーミッションを持つ 1 つ以上のチームメンバーである必要があります。

- ※ ドメイン—プロジェクト (表示および編集)

1.1.2. 新規プロジェクト

プロジェクトは、異なるプラットフォーム用のアプリとそれらが通信するクラウドアプリで構成されます。



注記

新規プロジェクトは通常、クライアントアプリ 1 つとその通信先となるクラウドアプリで構成されますが、プロジェクトには複数のアプリを追加することができます。

1.1.2.1. クライアントのプラットフォーム

プロジェクトにはすべて、出発点としてテンプレートが用意されています。以下の例ではテンプレートの構成を説明しています。

- ※ 単一アプリで、これが Node.js を使用する Cordova とクラウドアプリを使用する。

1.1.3. インポート

アプリ作成に加えて、既存のアプリをプラットフォームにインポートすることもできます。アプリをインポートする際には、新規プロジェクトを作成してそこにアプリをインポートするか、既存のプロジェクトにアプリをインポートすることを選択できます。プラットフォームにインポート可能なアプリのタイプは数多くあり、Native iOS、Native Android、PhoneGap、Basic Web App、および Advanced Web App などがあります。また、アプリをプラットフォームにインポートするには、以下のような方法があります。

- ※ 既存の Git レポジトリをクローンする
- ※ Zip をアップロードする
- ※ 空のレポジトリでアプリを作成し、そこにコードをプッシュする
- ※ ソースをインポートせずに RHMAP SDK を統合する

アプリをプラットフォームにインポートする方法については、ユーザーにステップごとの指示が示されます。この方法は、選択するインポートのメソッドによって多少異なります。例えば、「既存の Git レポジトリをクローンする」オプションでのみ、Git レポジトリのクローン方法に関する指示があります。

1.1.4. アプリ、クラウドアプリ & サービス

アプリ、クラウドアプリ & サービスのセクションでは、プロジェクト内のクライアントアプリとクラウドアプリの管理ができます。ここでは、新規クライアントアプリとクラウドアプリをプロジェクトに追加することができます。クラウドアプリは、クライアントアプリとクラウドバックエンドのストレージおよびサービスをつなぐゲートウェイの役割を果たします。

1.1.4.1. アプリ

プロジェクトは複数のアプリで構成することができます。一般的には、クライアントアプリ 1 つとその通信先となるクラウドアプリ 1 つで構成されます。プロジェクトには複数のクライアントアプリとクラウドアプリを追加することが可能です。クライアントアプリは、数多くの Native や Hybrid テンプレートをベースにして作成してから追加することができます。また、上記にあるように既存のアプリをインポートすることもできます。

1.1.4.2. クラウドコードアプリ

クラウドアプリを使用すると、開発者はアプリをバックエンドのクラウドストレージにリンクすることができます。これにより幅広いユーザー管理が可能になり、ユーザーはオンラインデータストレージや web ベースの E メールサービス、プッシュ通知などの MBaaS サービスを管理できるようになります。例えば、クラウド内で作成した機能はすべて簡単な API コールでアクセスできるので、パフォーマンス目的でクラウドキャッシュを活用してデバイス自体の作業を減らすことができます。

1.1.4.3. サービス

外部の MBaaS サービスをプロジェクトに追加して機能を拡張することができます。サービスをプロジェクトに追加する方法は、クライアントアプリや MBaaS クラウドインスタンスの場合と同様です。クラウドインスタンスはこれらのサービスを呼び出し、PayPal などの機能を活用することができます。MBaaS サービスのプロビジョニングができるのは、[サービス管理者](#)のロールがあるユーザーのみです。サービス管理者は、サービス作成時にこれをグローバルに全プロジェクトで利用可能とするオプションがあります。グローバルで利用可能とした場合は、[開発者](#)のロールを持つユーザーは、これを選択してプロジェクトに含めることができます。

1.1.5. アプリをプロジェクトに追加する

プロジェクトにはいくつでもアプリを追加することができ、以下の 3 つの方法があります。

1.1.5.1. 新規アプリの作成

新規アプリの作成では、アプリのベースとなるテンプレートを選択できます。既存のアプリがある場合は、「既存のアプリをインポート」のオプションを選択できます。

1.1.5.2. 既存アプリのインポート

既存のアプリがある場合は、このオプションを選択してご使用のプラットフォームと統合することができます。適切なアプリのタイプを選択して、前提条件が満たされていることを確認します。App Studio がこのプロセスをサポートします。

1.1.5.3. 既存アプリのクローン

ここに (または別のプロジェクトに) クローンをしたいアプリがある場合は、このオプションを選択します。クローンしたアプリには新しい名前を付けることが求められます。ソースとなるアプリによっては、クローンに関する新たな選択肢が表示される場合もあります。

アプリによってはクローン中にタイプが変更されるものもあれば、既存のアプリと Git リポジトリが共有できるものもあります。アプリには専用の Git リポジトリがあることが推奨されますが、アプリ間で Git リポジトリを共有することが望ましいインスタンスもあります (例: Basic Web App と Cordova App が非常によく似たソースを共有する場合)。このようなケースでは、アプリはリポジトリを共有し、共有レポジトリ内にフォルダーを作成し、それをビルド、レビュー、およびデプロイに使用できます。

1.1.6. 接続

接続セクションでは、選択したプロジェクトで構築されたバイナリーの概要と実行中のクラウドアプリの概要が提供されます。プロジェクトにはクライアントアプリとクラウドアプリの両方が含まれており、接続はクラウドアプリアプリがポイントするクラウドインスタンスを決定します。接続は、環境内の特定のクラウドアプリをポイントします。ダッシュボードでは、全環境内に存在する接続の概要が示されます。ユーザーはダッシュボードで、各クライアントアプリがどのようにクラウドインスタンスと接続しているかを見ることができます。クライアントアプリとクラウドアプリとの接続は、何回でも再設定したり無効にすることができます。

1.1.6.1. SDK Config

クライアントアプリとクラウドインスタンス間で接続が作成されると、SDK 設定ファイルが生成されます。**SDK Config** オプションを選択すると、この設定ファイルが表示されます。このファイルのコンテンツを自分のクライアントアプリの **fhconfig.json** ファイルにコピーすると、クライアントアプリにこの特定の接続を使用するように指示できます。

1.1.6.2. 接続の再設定

接続は再設定して、クライアントアプリが異なるクラウドインスタンスと通信できるようにできます。クラウド接続は、アプリの起動時にリフレッシュされます。

1.1.6.3. 接続を無効にする

これにより、クライアントアプリとクラウドアプリ間の接続が削除されます。プロジェクトに新規アプリが追加され、更新が必要な場合にこれを行う必要があります。接続を無効にすると、クライアントアプリは機能しなくなります。クライアントに接続が無効化されるという情報を送信することができます。

1.1.7. Git クイックスタート

Git クイックスタートページでは、Git を使って自分のローカルシステム上のアプリのクライアントおよびクラウドコードにアクセスする方法を示しています。リポジトリをクローンすると、既存リポジトリと全く同じコピーが作成されます。'git clone' を実行すると、プロジェクト内の全ファイルの全バージョンがプルされます。ただし、リポジトリをクローンする前に自分の SSH パブリックキーをアップロードする必要があります。パブリックキーがアップロードされたら、リポジトリは自由にクローンすることができます。

Git リポジトリのクローンについての詳細情報は、Git ドキュメンテーションの [Getting started with Git](#) を参照してください。

便利な Git コマンド一覧は、[Git Reference](#) を参照してください。

1.1.8. リソース

リソースセクションでは、全環境にまたがるプロジェクトのアプリについて、CPU やメモリー使用量 & ディスク使用率などのリソース使用率が表示されます。特定の環境にアプリがデプロイされていない場合は、この環境でのリソース使用率は表示されません。この画面では、デプロイ済みアプリのステータスを管理することもできます。例えば、特定のアプリのメモリー使用率が高い場合は、これを停止することができます。リソースのグラフはライブのデータを基にしているため、データを定期的に収集する間、傾向を表示するまでに多少時間がかかる場合もあります。

1.1.9. ライフサイクル管理

プロジェクトは通常、単一のアプリ (例: Cordova App) とクラウドアプリ (例: Node.js Application) で開始されます。プロジェクトは進捗するにつれて様々なステージを経過します。これがプロジェクトのライフサイクルになります。プロジェクトはそれぞれ異なるペースで進行し、その構成も時間の経過とともに拡大し、異なってきます。ライフサイクル管理ダッシュボードは、開発者とプロジェクト管理者の両方にとってこのプロセスを容易なものにします。

ダッシュボードでは、その時点におけるプロジェクトのスナップショットを提供します。アプリはすべて、ダッシュボード上の行で示されます。各環境は、ダッシュボードの列で示されます。このグリッドにより、プロジェクトの各部分がどのステージにあるか、個別のチームメンバーに質問することなく概要がすぐに分かります。

アプリがビルドされていれば、クラウドアプリがあるターゲットとなっている環境を示す列に直近のビルドが表示されます。アプリのバイナリーはここからダウンロードが可能で、または異なる git ブランチやタグから再ビルドすることもできます。「プロモート」オプションは同一コードの再ビルドのショートカットになりますが、別のクラウドアプリ環境に接続します。

クラウドアプリがデプロイ済みの場合は、直近のデプロイの詳細とステータスが対応する環境列に表示されます。クラウドアプリのコードは、利用可能な環境であればいつでもデプロイ (または再デプロイ) することが可能です。クラウドアプリの「プロモート」オプションは、既にデプロイ済みの git ブランチまたはタグを別の環境にデプロイするためのショートカットです。



注記

プラットフォーム外で実行されたアプリのバイナリービルドは、ライフサイクル管理ダッシュボードでは反映されません。

1.1.10. フォーム

フォーム設定ページでは、ユーザーはプロジェクト内でフォームとテーマを関連付け、クライアント側のアプリフォームを設定することができます。ユーザーは、カメラ設定や提出、管理ユーザー、クライアントログイン、およびクラウドログインを管理することができます。

プロジェクトに関連付けられている提出もこのページで見ることができます。



注記

ユーザーに十分な権限がない場合は、提出を見ることができません。ユーザーのロールに関する詳細情報は、[管理に関するドキュメント](#)を参照してください。

1.1.11. レポート



注記

レポートデータは、1 日に 1 度生成されます。

レポートのセクションでは、特定のプロジェクト内のクライアントとクラウドアプリの両方に関するアナリティクス情報が高度なグラフで提示されます。実行されたデバイスインストール数や起動したアプリ数など、幅広い情報にユーザーはアクセスできます。これらの分野はどれでもさらに調査して、日付やプラットフォーム、場所に基づくアナリティクス情報にアクセスすることができます。

Studio では、これらの情報は多くの異なるグラフで表示されます。日付に特定したデータは折れ線グラフで示されます。これによりユーザーは、特定期間のアプリのアクティビティーの内訳を知ることができます。範囲内の特定日におけるアクティビティーの量が分かります。例えば、範囲内の特定日に実行されたクラウドリクエストの数が確認できます。

ドーナツグラフでは、プラットフォームごとのアクティビティーの内訳が表示されます。例えば、特定の期間内でプラットフォームごとに実行されたインストール数などです。場所特定のデータは、地図で示されます。これにより、ユーザーは地理的位置に基づいたアプリのアクティビティー内訳が分かります。各グラフには対応する凡例があります。カーソルをこれらの図表上に持って行くと凡例で対応する値の詳細情報が示されます。

データ収集を行う期間は変更することができます。事前定義の期間では、「過去 7 日間」「過去 30 日間」「前月」の情報が示されますが、これらのいずれも当てはまらない場合は、範囲の開始日と終了日を手動で選択することもできます。つまりユーザーはデータの 카테고리を選択し、日付の範囲を入力するだけでアプリのデータを入手することができます。

1.1.12. 設定

プロジェクト設定ページから、プロジェクトの名前を変更したり削除することができます。

1.1.12.1. プロジェクトの削除

プロジェクトは一旦削除したら、元に戻すことはできません。Git リポジトリ、クライアントアプリ、およびクラウドアプリのすべてが削除されます。削除されるプロジェクトに関連付けられているアプリは機能しなくなります。

1.1.12.2. プロジェクトの名前変更

ここではプロジェクトの名前が変更できます。

1.2. アプリ

1.2.1. 概要

Red Hat Mobile Application Platform ホスト型 (RHMAP) を使うと、hybrid、HTML5、または Native Apps など幅広いアプリをビルドできます。クライアントアプリとクラウドアプリは両方ともプロジェクトに追加できます。アプリは、位置情報やバックエンドストレージなどの機能へのアクセスを提供する数多くの JavaScript API にアクセスがあります。フォームベースのアプリは、インターフェースの簡単なドラッグアンドドロップで作成できます。

要件

ユーザーは、以下のパーミッションを持つ 1 つ以上のチームメンバーである必要があります。

- ✦ ドメイン — プロジェクト (表示)
- ✦ クライアントアプリ (表示 & 編集) — アナリティクス (表示)

関連リソース

- ✦ [FHC CLI](#)

1.2.2. 詳細

アプリ、クラウドアプリ & サービス画面を選択すると、デフォルトで詳細ページが表示されます。ここでは、現在選択しているアプリに関する以下の情報の概要が示されます。

- ✦ アプリのタイプ: 使用するアプリのタイプです。
- ✦ アプリの名前: 使用するアプリの名前です。クライアントアプリの場合は、これがデバイスで表示されるアプリの名前になります。
- ✦ アプリの説明: 使用するアプリの説明です。
- ✦ アプリ ID: アプリの一意の識別子です。このアプリを FHC で用いてアクションを実行するには、これを使用します。
- ✦ アプリの説明: 使用するアプリの説明です。
- ✦ アプリ API キー: これをアプリ ID と合わせて使用することで、アプリが選択したクラウドアプリと通信できるようになります。
- ✦ ソースパス (Cordova のみ): アプリの全ソースファイルを格納しているフォルダーです。

これは通常、**www** になります。アプリのビルドまたはデプロイ時に別のフォルダーを使用したい場合は、ここで変更します。前にあるまたは後ろに続くスラッシュはパスから除かれます。例えば、ソースフォルダーが **/mysource/folder/** の場合、入力するパスは **mysource/folder** とします。アプリにおけるビルドプロセス (Grunt 縮小など) は、最終的なアセットをこのフォルダーに格納します。
- ✦ Git SSH URL: リポジトリをクローンする URL です。このリポジトリを他のアプリと共有する場合、それらのアプリはここに記載されます。
- ✦ Git ブランチ: Git リポジトリから checkout するブランチです。

いずれのアプリの詳細もこのページで更新できます。Studio では、この画面にアプリのプレビューもあります。ユーザーは、アプリを電話、タブレット、またはデスクトップでプレビューすることが選択できます。アプリプレビューウィンドウでデバイスを選択したら、ドロップダウンリストから特定のデバイスを選択してアプリのプレビューを実行することができます。

アプリは、アプリ、クラウドアプリ & サービスページから削除することもできます。



注記

アプリを削除すると関連する Git リポジトリも削除され、クラウドアプリの場合はデプロイ解除されます。

1.2.3. エディター

ビルトインのエディターを使うことで、アプリ内のコードが編集できます。構文ハイライト表示の機能が組み込まれているため、構文要素を区別しやすくなっています。ユーザーはファイルやフォルダーを新規作成し、保存された変更は自動的に対応する Git リポジトリにプッシュバックされます。

変更がなされたらプレビューが自動的にリフレッシュされるため、変更の反映がすぐに確認できます。'git pull' を実行すると、Git リポジトリから変更がプルされます。以下の Git 機能は、Studio のエディターセクションで利用できます。

- ※ **branch** - 詳細は [Git Branching - Branches in a Nutshell](#) か **git-branch** API を参照してください。
- ※ **tag** - 詳細は [Git Basics - Tagging](#) か **git-tag** API を参照してください。
- ※ **commit** - 詳細は [Git Basics - Recording Changes to the Repository](#) の **Committing Your Changes** セクションか **git-commit** API を参照してください。
- ※ **clone** - 詳細は [Git Basics - Getting a Git Repository](#) の **Cloning an Existing Repository** セクションか **git-clone** API を参照してください。
- ※ **pull** - 詳細は **git-pull** API を参照してください。

エディターのアプリプレビューでは、多くの選択可能なデバイスで現行アプリがどのように表示されるかを確認できます。

1.2.4. Docs

アプリに **README.md** のマークダウンファイルが含まれる場合は、変換されてここに表示されます。

1.2.5. アナリティクス



注記

アナリティクスデータは、1 日に 1 度生成されます。

アナリティクスセクションでは、アプリ使用量統計値のグラフィック表示にアクセスできます。データは、インストールやクラウドリクエスト、起動、アクティブなデバイスなど多くのカテゴリーで調べることができます。また、日付やプラットフォーム、場所でカテゴリー分けすることもできます。グラフのコンテンツは、「場所ごとのインストール」を除いてプラットフォーム別に表示されます。各プラットフォームは、凡例でそれぞれ別のキーで示されます。グラフ内のコンテンツは、凡例で各プラットフォームをクリックするとオンとオフが切り替えられます。

日付の範囲もカスタマイズが可能です。事前定義の期間は、「過去 7 日間」「過去 30 日間」「前月」ですが、独自の開始日と終了日を適用することもできます。



注記

アナリティクスページは「レポート」タブとは異なります。レポートタブではプロジェクトごとのアナリティクス情報が表示されます。アナリティクスページでは、アプリごとのデータが表示されます。

アプリの初回起動時にクラウドにコールバックが行われ、この時点でデバイスインストールが記録されます。デバイスインストールはユーザーインストールとは異なります。ユーザーインストールは、以下の方法で App Stores が追跡するものです。

- ※ アプリストアは、ユーザーがアプリをインストールしたデバイス数にかかわらず、インストールをユーザーアカウントのダウンロード実行として数えます。
- ※ アプリストアは通常 PST タイムゾーンで日次結果をレポートしますが、RHMAPレポートは UTC に準拠しています。
- ※ RHMAP がインストールを追跡するには、アプリが有効なインターネット接続で少なくとも 1 回は起動される必要があります。ここでインストールがカウントされるのは、ダウンロード時ではなく、アプリの初回起動時です。

1.2.6. プッシュ

プッシュセクションでは、アプリのプッシュ通知の設定ができます。プッシュ通知についての詳細は、[プッシュ通知](#) の章を参照してください。

1.2.7. ビルド

ビルド機能を使うと、選択したプラットフォーム向けにバイナリーのセットを構築できます。ビルド操作を実行する際には、バイナリーの接続先となるクラウドアプリを選択する必要があります。これは、プロジェクトと関連付けられているクラウドアプリであれば、どれでも構いません。ビルドは以下のプラットフォーム向けに実行できます。

- ※ Android
- ※ iPhone
- ※ iPad
- ※ iOS Universal

また、以下のいずれかのビルドタイプを選択する必要があります。

- ※ 開発ビルド
- ※ 配信ビルド
- ※ リリースビルド
- ※ デバッグビルド

選択したプラットフォームとビルドタイプによっては、バイナリーの署名に認証情報バンドルが必要になる場合があります。認証情報については、下記の証明書セクションを参照してください。

ビルドが成功したら、そのアーティファクトがアーティファクト履歴に追加されます。アーティファクト履歴では、バイナリーが構築されたプラットフォームやビルドタイプ、ビルドに使用された認証情報バンドルなどを含むビルドの概要情報が提供されます。アーティファクト履歴からアーティファクトを選択すると、そのバイナリーを再度ダウンロードすることができます。



注記

ビルドを実行するには、まず認証情報ページから関連する認証情報をアップロードする必要があります。

1.2.7.1. アプリバイナリーをサードパーティー MAM および MDM プロバイダーに公開する

RHMAP では、サードパーティー MAM および MDM プロバイダーへのアプリバイナリーのアップロードに対応しています。個別プロバイダーのサポートは、**管理 > モバイルアプリの管理 > サードパーティー MAM/MDM** セクションで有効、無効にできます。

少なくとも 1 つの MAM または MDM プロバイダーが有効になっていれば、クラウドアプリの **ビルド** 画面の **MDM 統合** セクションで個別プロバイダーのオプションが表示されます。

1.2.7.1.1. Apperian

生成したバイナリーがビルド後に Apperian App Catalog にアップロードされるようにするには、ビルド前に以下のステップを実行します。

1. RHMAP インスタンスのドメイン名が Apperian の EASE Portal にあるホワイトリストに追加されていることを確認します。詳細は、公式 Apperian ドキュメントの [Enable a Domain Whitelist for Custom Admin Sites](#) を参照してください。
2. アプリバイナリーが Apperian App Catalog にある要件を満たしていることを確認します。要件の完全一覧については、公式 Apperian ドキュメントの [Add an Application](#) を参照してください。



注記

RHMAP クライアントアプリのテンプレートには、要件のすべてを満たしていないものもあります。例えば、アプリのアイコンファイルの設定が必要な場合があります。

3. 特定のビルドで Apperian への公開を有効にする。
 - a. **Push the generated app binary to the Apperian App Catalog** を選択します。
 - b. Apperian ユーザー名とパスワードを入力します。
 - c. **Apperian Login** をクリックします。認証が成功すると、**OAuth Token** フィールドに Apperian API との通信に使用されるトークンが表示されます。

これ以降のビルドにおけるアプリバイナリーは、Apperian App Catalog にアップロードされます。

1.2.8. エクスポート

エクスポートセクションでは、アプリをいくつかの選択したプラットフォームでエクスポートできます。アプリをエクスポートする際には、バイナリーの接続先となるクラウドアプリ、プラットフォーム、およびプラットフォームのバージョンを指定する必要があります。アプリは zip ファイルとしてエクスポートされます。エクスポートが完了すると、エクスポートはアーティファクト履歴に追加され、ここからいつでも再ダウンロードできるようになります。

1.2.9. 認証情報

認証情報セクションでは、認証情報バンドルが管理できます。新規の認証情報バンドルは追加またはインポートが可能です。また、既存の認証情報バンドルを一覧表示することもできます。



注記

アプリをビルドするには、関連する認証情報バンドルへのアクセスを読み込んでいる必要があります。このアクセスを読み込んでいない場合は、ビルドは失敗します。

1.2.9.1. 認証情報バンドル

特定のタイプのビルド実行が可能となるには、アプリが認証情報バンドルで署名されている必要があります。認証情報バンドルは、開発ビルド、配信ビルド、デバッグビルドなど特定のタイプのビルドに必要な証明書やプロビジョニングプロファイル、およびプライベートキーなどのリソースの組み合わせです。プラットフォームおよびビルドタイプによって、バンドルは異なるリソースの組み合わせになります。認証情報バンドルを構成する個別リソースについての詳細情報は、[認証情報バンドルのコンポーネント](#) を参照してください。



注記

iOS 向けのアプリを開発するには、Apple Developer Account が必要になります。

各プラットフォームのビルドタイプで必要となる認証情報は、以下の通りです。



注記

プライベートキーとそのパスワードが安全な場所に保存されていることを確認してください。これらは、RHMMap で認証情報バンドルの設定時に必要となります。証明書の発行時には、正しいプライベートキーが使用されていることを確認してください。たとえば、iOS 認証情報バンドルで作業する際には、開発ビルドのプライベートキーを使用して開発ビルドの開発者証明書を生成するようにします。

iOS 認証情報バンドル

- ※ 開発ビルド
 - 開発者証明書
 - プライベートキー
 - 開発プロビジョニングプロファイル
- ※ 配信ビルド

- 配信証明書
- プライベートキー
- 配信プロビジョニングプロファイル
- ※ リリースビルド
 - 配信証明書
 - プライベートキー
 - 配信プロビジョニングプロファイル

Android 認証情報バンドル

- ※ デバッグビルド
 - なし
- ※ リリースビルド
 - 証明書
 - プライベートキー

1.2.9.1.1. 新規バンドルの作成

ここでは新規の認証情報バンドルを作成し、後でビルド中にバイナリーの署名にこれを使用することができます。選択したプラットフォームによっては、バンドルに異なる認証情報を追加する必要があります。例えば、Android 認証情報バンドルの作成にはプライベートキーと証明書が必要になりますが、iOS 認証情報バンドル作成時には、プライベートキーと証明書、さらにプロビジョニングプロファイルが必要になります。

1.2.9.1.2. 既存の開発リソースの移行

これにより古い App Studio から既存の認証情報が移行され、認証情報バンドル一覧に追加されます。

新規バンドルが作成されるか、移行でインポートされると、認証情報バンドル一覧に表示されます。

1.2.10. 統合

統合セクションでは、既存のクライアントアプリを RHMAP プロジェクトにインポートすることができます。アプリをインポートするには、レポジトリをクローンして、アプリを RHMAP SDK と統合する必要があります。

1. Git レポジトリのクローン

リポジトリをクローンすると、既存リポジトリと全く同じコピーが作成されます。'git clone' を実行すると、プロジェクト内の全ファイルの全バージョンがプルされます。ただし、リポジトリをクローンする前に自分の SSH パブリックキーをアップロードする必要があります。パブリックキーがアップロードされたら、リポジトリは自由にクローンすることができます。

Git リポジトリのクローンについての詳細情報は、Git ドキュメンテーションの [Getting started with Git](#) を参照してください。

便利な Git コマンド一覧は、[Git Reference](#) を参照してください。

2. SDK のダウンロード

アプリが機能するには、アプリを RHMAP SDK と統合する必要があります。最新の RHMAP JavaScript SDK をダウンロードするためのリンクが画面に表示されます。

3. JavaScript ファイルをプロジェクトにコピーする

RHMAP SDK をダウンロードしたら、作成したプロジェクトにこれを移動して参照することができます。

4. 初期化スニペットのコピー

アプリの ID、アプリキー、ホストドメイン、関連付けるプロジェクトの ID を格納する **fhconfig.json** ファイルを作成します。

5. Git Commit および Push

SDK とアプリの統合が成功したら、変更のコミットとプッシュを実行できます。これでアプリがプロジェクトと正常に関連付けられました。

1.3. クラウドアプリ

1.3.1. 概要

クラウドアプリはサーバー側のアプリケーションで、クラウドの実行環境である **MBaaS** で稼働します。クライアントアプリはクラウドアプリと通信し、データストレージや認証、キャッシングなどのバックエンド機能にアクセスします。

要件

ユーザーは、以下のパーミッションを持つ 1 つ以上のチームメンバーである必要があります。

- ※ ドメインプロジェクト (表示)
- ※ クラウドアプリ (表示 & 編集) — アナリティクス (表示)

関連リソース

- ※ [クラウド API リファレンス - クラウド API コール一覧](#)
- ※ [FHC CLI](#)
- ※ [プラットフォーム API - クラウド警告 API](#)
- ※ [プラットフォーム API - クラウドイベント API](#)

1.3.2. 詳細

詳細のセクションでは、クラウドアプリの概要情報にアクセスできます。このセクションでは、このデータを編集、管理することができます。

アプリは、Running または Stopped のいずれかのステータスになり、ステータスは **Start**

App、**Stop App**、および **Restart App** ボタンで制御されます。アプリが Running 状態の場合は、そのアプリのデバイス上のビルドは予想どおりに機能します。ただし、Stopped ステータスの場合は、アプリがクラウドの機能に依存していれば、デバイス上のユーザーはそのアプリを使用することができません。

また、アプリ ID、アプリ API キー、名前などの一般的な情報も記載されています。これらの情報は、このセクションで編集および更新ができます。

- ※ アプリ ID: アプリの一意の識別子です。
- ※ 名前: クラウドインスタンスの名前です。
- ※ 説明: クラウドインスタンスの簡単な説明です。
- ※ アプリ API キー: これをアプリ ID と合わせて使用することで、アプリがクラウドと通信できるようになります。
- ※ Git URL: リポジトリをクローンする URL です。
- ※ Git ブランチ: Git リポジトリから checkout するブランチです。

1.3.3. エディター

ビルトインのエディターを使うことで、クラウドアプリ内のコードが編集できます。構文ハイライト表示の機能が組み込まれているため、構文要素を区別しやすくなっています。ユーザーはファイルやフォルダーを新規作成し、保存された変更は自動的に対応する Git リポジトリにプッシュバックされます。

変更がなされたらプレビューが自動的にリフレッシュされるため、変更の反映がすぐに確認できます。**git pull** を実行すると、Git リポジトリから変更がプルされます。

エディターのアプリプレビューでは、多くの選択可能なデバイスで現行アプリがどのように表示されるかを確認できます。

1.3.4. 環境変数

Studio を使用する環境変数は定義が可能で、これらの変数はクラウドコードでアクセスできます。Studio では、各クラウド環境における現在定義済みの環境変数とデプロイ済みの環境変数の一覧を表示します。また、システムが定義する環境変数一覧も見ることができます。

1.3.4.1. 環境変数の一覧

環境変数のオプションは、左側のメニューにあります。最初の表示では、現在のクラウドインスタンスの環境変数全一覧が表示されます。

1.3.4.1.1. アプリの環境変数

このセクションでは、ユーザーが定義した変数が一覧表示されます。これらの変数はクラウドにプッシュされていないことがあるので、値は現在アクティブなものとは異なる場合があります。定義済みの値とデプロイ済みの値で不整合がある場合は、赤字で表示されます。値のフィールドをクリックすると、これを更新することができます。

1.3.4.1.2. デプロイ済みユーザーの環境変数

このセクションでは、ユーザーが作成したデプロイ済みの環境変数が一覧表示されます。上記と同様に、不整合は赤字で表示されます。ただし、これらの値は直接は更新できません。

1.3.4.1.3. デプロイ済みのシステム環境変数

このセクションでは、プラットフォームが作成したデプロイ済みの環境変数が一覧表示されます。これらの値はリファレンス目的のみとなり、これら変数の値は変更しないでください。

1.3.4.2. 環境変数の作成

新規の環境変数は、**変数を追加** ボタンをクリックすると作成できます。環境変数の名前は以下のルールにしたがってください。

※ **FH_** または **MONIT_** で始めない。

※ 以下のいずれも使用しない。

- **PATH**
- **HOME**
- **PWD**
- **USER**
- **NODE_PATH**
- **LD_LIBRARY_PATH**

1.3.4.3. 環境変数の設定解除/削除

既存の環境変数一覧で変数にチェックを入れると、変数を設定解除または削除するオプションがあります。

※ **設定解除** 選択された環境から変数が削除されます。

※ **削除** すべての環境から変数が削除されます。

同時に複数の変数を設定解除または削除することが可能です。

1.3.4.4. 環境変数をプッシュ

変数の値の更新が完了したら、「環境変数をプッシュ」のボタンをクリックして変数をクラウドにプッシュする必要があります。これを実行すると、アプリの再起動も実行されます。

1.3.4.5. 環境変数のコードでの参照

環境変数は、`process.env.variable` 名を使用することで直接クラウドコード内で参照することができます。例を示します。

```
// PIN_ENABLED is the environmental variable created in the Studio.
var pin_enabled = process.env.PIN_ENABLED || true // defaults to true
if no value defined
```

クライアント側から変数の値を取得したい場合は、変数の値を返すクラウド側の関数をエクスポートし、クライアント側からこの関数を呼び出します。

```
// On the cloud side using node.js
exports.getEnvVariable = function(params, cb){
```

```

    var pin_enabled = process.env.PIN_ENABLED || true

    return cb(null, {
      enabled : pin_enabled
    });
  };

  // On the client side
  $fh.act({
    act : 'getEnvVariable',
    req : {}
  }, function(res){
    // run this in the event of success.
  }, function(err){
    // run this in the event of failure.
  });

```

1.3.5. データブラウザー

App Studio データブラウザーセクションでは、開発者は以下のことができます。

- ※ アプリに関連するデータを視覚的かつ対話的に表示する。
- ※ コレクションを表示、作成、削除、インポート、エクスポートする。
- ※ コレクション内のデータを修正する。

詳細情報は [データブラウザーガイド](#) を参照してください。

1.3.6. デプロイ

デプロイセクションでは、開発者は以下のことができます。

- ※ 現行アプリのクラウドコードをデプロイし、デプロイのログを表示する。
- ※ クラウドコードをデプロイする環境 (複数利用可の場合) を選択する。
- ※ デプロイする GIT ブランチまたはタグを選択する (利用可能な場合)。
- ※ クラウドコードを実行するランタイム & バージョンを選択する (利用可能な場合)。
- ※ クリーンデプロイを実行する。ここでは、(依存関係を含む) デプロイディレクトリー全体が削除され、最初から再デプロイが実行されます。
- ※ 選択したブランチに、今後の Git プッシュで自動的にアプリがデプロイされるかどうかを選択する。

Node.js アプリケーションのデプロイプロセスは通常、以下のようになります。

- ※ 指定されたブランチまたはタグリファレンスでクラウドコードを取得する。
- ※ (内部およびカスタムの) 環境変数の値すべて、および選択した環境のその他の設定 (例: ランタイムなど) を収集する。
- ※ クラウドコード、環境変数の値、および他の設定を選択した環境にプッシュする。
- ※ クラウドコードに関連するデプロイフォルダーに展開/コピーする。

- ※ 環境変数がアプリの起動スクリプトで設定される。
- ※ 起動スクリプトで正常に設定されているランタイムなど、送信される設定内容が実行される。
- ※ 依存関係が解決され、インストールされる (package.json から)。
- ※ 起動スクリプトを使用してアプリが起動する。

上記のステップは、選択したオプションやアプリのタイプ、特定ファイルの有無およびそのコンテンツによって異なります。

さらに開発者は、クラウドコードを OpenShift Online にデプロイするようプラットフォームを設定することができます。詳細は、[Staging Cloud Apps to RedHat OpenShift Online PaaS guide](#) を参照してください。

1.3.7. 統計

Studio では、アプリのエンドポイント統計値とカスタム統計値という 2 種類の統計があります。これらは、クラウドアプリの統計セクションにあります。

これらの統計数値を表すグラフには共通の UI 機能があります。データは異なる形式で (例: csv) ダウンロードできます。データの範囲は、各グラフの下にあるスクロールバーとスライダーで変更可能です。グラフ内の各行は、凡例のアイテムをクリックすることでオンとオフが切り替えられます。

1.3.7.1. アプリのエンドポイント

アプリのエンドポイント統計値はすべてのアプリで追加設定なしで使用できます。**main.js** で定義されているエンドポイントは、エンドポイントが呼び出されると統計値の生成を開始します。「すべてのエンドポイント」の結合グラフも統計値が生成されると利用可能になります。

統計値を生成している各エンドポイントでは、以下のサマリー情報が利用可能となります。

- ※ 1 分あたりのリクエスト数 - 直近 1 分間のリクエスト数 (短い間隔を基に計算された値)
- ※ 平均リクエスト時間 - 直近の間隔でのリクエストあたりの平均時間
- ※ 平均同時リクエスト - 直近の間隔で処理された同時リクエストの平均数 (間隔あたりのリクエスト数と間隔あたりの平均リクエスト時間を基に計算)

デフォルトのグラフビューでは以下のデータが表示されます。

- ※ 1 秒あたりのリクエスト数 - アプリが 1 秒間に処理したリクエスト数 (直近の間隔を基に計算)
- ※ 平均同時リクエスト - 直近の間隔で処理された同時リクエストの平均数 (間隔あたりのリクエスト数と間隔あたりの平均リクエスト時間を基に計算)
- ※ 平均リクエスト時間 (90 パーセンタイル) - 直近の間隔でのリクエストの平均時間。外れ値は無視する。

グラフ上部の凡例にある名前をクリックすると、最長リクエスト時間や最短リクエスト時間といった他のグラフアイテムも表示できます。その下のグラフでは、それらの統計値を含めたものが表示され、1 秒あたりのリクエスト数と平均同時リクエスト数の選択は解除されます。

1.3.7.2. カスタム統計値

カスタム統計値は、[\\$fh.stats](#) API を使って開発者が定義します。

1.3.7.2.1. カウンター

カウンターは線グラフ上で見ることができ、各間隔のカウンター値が表示されます。



注記

カウンターは各間隔ごとにリセットされます。

1.3.7.2.2. タイマー

タイマーも線グラフ上で見ることができます。各タイマーの値は、該当間隔におけるそのタイマーの平均値 (90 パーセンタイル) になります。各間隔における上限値と下限値も別の線でグラフ化されます。これらの値により、平均時間においてタイマーがどのように変化したかについての概要が分かります。

1.3.8. 通知およびイベント

アプリでは、アプリに対して特定のアクションが発生すると、イベントが生成されます。これらのイベントはプラットフォームが記録し、App Studio と FHC が開発者に提示します。例えば、開発者がアプリをデプロイすると、プラットフォームはデプロイの発生時間とデプロイ実行者、およびその結果を記録します。

イベントにはカテゴリ、重大度、および名前があります。現在プラットフォームが生成するイベントは以下の通りです。

イベント名	イベントカテゴリ	イベント重大度	説明
CREATE_REQUESTED	APP_STATE	INFO	アプリ作成がリクエストされます。
CREATED	APP_STATE	INFO	アプリが作成されます。
CREATE_FAILED	APP_STATE	ERROR	アプリの作成に失敗しました。
DEPLOY_REQUESTED	APP_STATE	INFO	アプリのデプロイがリクエストされます。
DEPLOYED	APP_STATE	INFO	アプリがデプロイされます。

イベント名	イベントカテゴリ	イベント重大度	説明
DEPLOY_FAILED	APP_STATE	ERROR	アプリのデプロイが失敗しました。
START_REQUESTED	APP_STATE	INFO	アプリの起動がリクエストされます。
START_SUCCESSFUL	APP_STATE	INFO	アプリが起動しました。
START_FAILED	APP_STATE	ERROR	アプリの起動が失敗しました。
SUSPEND_SUCCESSFUL	APP_STATE	INFO	クラウドアプリへの HTTP コールが一定期間 (通常は 1 週間) ないというアクティビティがない状態になると、クラウドアプリが停止されます。クラウドアプリへの次の HTTP コールで自動的にアプリが再起動します。
STOP_REQUESTED	APP_STATE	INFO	アプリの停止がリクエストされます。
STOP_SUCCESSFUL	APP_STATE	INFO	アプリが停止されます。
STOP_FAILED	APP_STATE	ERROR	アプリの停止が失敗しました。
CRASHED	APP_STATE	ERROR	不明な例外がアプリから送られ、停止されました。システム監視プロセスにより自動的に再起動されます。

イベント名	イベントカテゴリ	イベント重大度	説明
KILLED_RESTARTED	APP_STATE	ERROR	アプリが停止し、再起動します。
TERMINATED	APP_STATE	FATAL	短時間に再起動が過剰に実行されたため (現在は 20 秒間に 10 回の再起動) アプリがシステムにより終了されました。システム監視プロセスが再起動することはありません。
DELETE_REQUESTED	APP_STATE	INFO	アプリの削除がリクエストされます。
DELETED	APP_STATE	INFO	アプリが削除されました。
DELETE_FAILED	APP_STATE	ERROR	アプリの削除に失敗しました。
CHANGE_REQUESTED	APP_ENVIRONMENT	INFO	環境変数の変更がリクエストされます。
CHANGE_SUCCESSFUL	APP_ENVIRONMENT	INFO	環境変数が変更されました。
CHANGE_FAILED	APP_ENVIRONMENT	ERROR	環境変数の変更失敗しました。

1.3.8.1. システムイベントビュー

プラットフォームで生成したイベントはすべて、Studio のクラウドアプリビューにある通知セクションで確認することができます。また、フィルターを使ってイベントを検索することもできます。イベントエントリーをクリックすると、イベントの詳細情報を見ることができます。

1.3.8.2. 警告 & 通知メール

特定のクラウドイベントが発生した際に開発者に通知メールを送信することができます。これは警告で実行されます。警告では、通知メールが実行される条件と使用する E メールアドレスを定義します。

1.3.8.2.1. 警告ビュー

警告ビューには、2つの表があります。1つ目では、該当アプリで作成された全イベントが一覧表示されます。2つ目では、選択された警告設定に合致するクラウドイベントすべてが表示されます。1つ目の表のエントリーをクリックすると、警告の詳細情報が示されます。

1.3.8.2.2. 警告の作成

「警告を作成」ボタンをクリックすると、警告作成ビューが表示されます。このビューでは、警告名、警告が監視するイベント、および警告の送信先となる E メールアドレスを指定します。

イベントを指定する際には、イベントカテゴリー、重大度、イベント名のいずれかまたはその組み合わせを使用します。すべてのフィールドで値を設定する必要はありません。警告が作成され、基準に合致するイベントが発生したら、警告で指定されたアドレスに通知メールが送信されます。

1.3.8.2.3. 通知ビュー

プラットフォームでは送信された通知メールすべての記録を維持し、これは Studio で見ることができます。表内のエントリーをクリックすると、通知の詳細が表示されます。

1.3.9. ログ

ユーザーは、各アプリ環境で作成された現行およびアーカイブ済みのアプリログにアクセスできます。

1.3.10. エンドポイントセキュリティ

エンドポイントセキュリティの機能を使用すると、アプリのエンドポイントに適用するセキュリティレベルを決定できます。

1.3.10.1. エンドポイントセキュリティの管理

エンドポイントセキュリティのオプションは、クラウドインスタンスの左側メニューにあります。セキュリティの設定には、以下の2つの方法があります。

1.3.10.1.1. アプリのセキュリティ

アプリのセキュリティでは、すべてのエンドポイントにわたって適用されるデフォルトのセキュリティレベルを定義します。デフォルトは HTTPS です。アプリ API キーオプションを適用すると、HTTPS は使用されますが、エンドポイントにアクセスするためには、アプリ API キーを送信し、アプリで作成したキーと一致する必要があります。このキーは、アプリの詳細メニューオプションにあります。デフォルトでは、このキーは RHMAP SDK から送信されます。このオプションを有効にすると、エンドポイントは正しい App API キーが送信された場合にのみ、アクセス可能になります。正しいキーがない場合は、401 (Not Authorised) が返され、エンドポイントは呼び出されません。

1.3.10.1.2. 個別のエンドポイントセキュリティ

アプリのエンドポイントに適用するセキュリティは、個別レベルで変更することが可能です。一覧を更新すると、クラウドインスタンスが再起動します。



注記

アプリが実行中でしばらくステージしていない場合は、最新バージョンのソフトウェアを取得するためにアプリを再度ステージする必要があります。

1.3.10.2. 監査ログ

アプリへの変更を透視的に見られるようにするために、セキュリティ設定が変更されると監査ログにエントリが作成されます。このログは、エンドポイントセキュリティ画面の上部にある監査ログをクリックすると表示されます。監査ログでは、使用中のアプリのセキュリティ設定で実行されたことのログが表示されます。フィルターを使って監査ログを絞り込むこともできます。

1.3.10.3. CORS サポート

CORS (クロスオリジンリソース共有) は、web ページ上の JavaScript がその JavaScript の元になっているドメインではなく、別のドメインに XMLHttpRequests を作成できるようにするメカニズムです。CORS についての詳細は、[CORS on Wikipedia](#) を参照してください。RHMAP では、全クラウドリクエストで自動的に CORS を有効にします。

選択したドメインへのアクセスは、以下のように制限できます。

- ※ Studio のクラウド管理セクションで「環境変数」をクリックします。
- ※ **CORS_WHITELIST** という名前の環境変数を追加します。
- ※ その値を、アクセスを制限するドメインに設定します。
- ※ 「環境変数をプッシュ」ボタンをクリックして変更を選択した環境に適用します。

1.4. MBAAS サービス

1.4.1. 概要

サービスとは、Oracle 統合サービスのようなバックエンドシステムと統合するためにプロジェクトが使用する稼働中のクラウドアプリです。サービスは 1 つ以上のプロジェクトに追加して、そのプロジェクトのアプリで利用可能とします。

要件

クラウドサービスのプロビジョニングを可能とするには、[サービス管理者](#) ロールをアカウントに割り当てる必要があります。

クラウドサービスのインスタンス作成を可能とするには、ユーザーのアカウントに [開発者管理](#) ロールか [開発者](#) が割り当てられ、プロジェクトの作成および管理が可能となっている必要があります。プロジェクトにアプリやクラウドインスタンスを追加した方法でサービスを追加することができます。

関連リソース

- ※ [Invoking a service via an API call](#)

※ Local Development with Services

1.4.2. MBaaS サービスのプロビジョニング

サービス管理者は、開発者による様々なバックエンドシステムへのアクセスを可能とする新規 MBaaS マイクロサービスのプロビジョニングを実行できます。サービスがプロビジョニングされたら、プラットフォーム上のプロジェクト内で使用可能になります。デフォルトでは、サービスが作成されるとプライベートとマークされます。ドメイン上の全プロジェクトでサービスを利用可能とするには、「グローバルサービス」オプションを有効にする必要があります。これでこのサービスが公開で利用可能であることが宣言されます。

1.4.3. 詳細

以下は特定のサービスに関連する概要情報になります。

1.4.3.1. サービスの詳細

サービスの詳細セクションでは、概要情報が提供されます。

- ※ サービス ID: 特定サービスの一意の識別子です。これは、FHC から特定サービスを参照する際に使用されます。
- ※ サービス API キー: これをアプリ ID と合わせて使用することで、アプリがクラウドと通信できるようになります。
- ※ Git URL: リポジトリをクローンする URL です。

1.4.3.2. サービス設定

サービス設定のセクションでは、サービスに関連する詳細が編集できます。

- ※ 名前: サービスの名前です。
- ※ プロジェクト: このサービスにアクセスがあるプロジェクト一覧です。
- ※ グローバルサービス: このオプションを選択すると、ドメイン上の全プロジェクトでサービスがグローバルに利用可能となります。この設定が有効であれば、開発者はこのサービスを自身のプロジェクト内で利用できます。
- ※ サービスの削除: サービスを削除します。

1.4.4. Docs

Docs セクションには API ドキュメンテーションビューワが含まれており、以下のような各エンドポイント向けのサービス API の重要な要素が明確に表示されます。

- ※ 期待されるリクエストヘッダーおよびリクエストボディの例。
- ※ サンプルリクエストの応答例。
- ※ 特定サービス上のメソッドを起動する **\$fh.service** コールの完全なスニペット。
- ※ ドキュメンテーションビューワから直接サービスを起動するフォーム。

[API Blueprint](#) 仕様に準ずる **README.md** ファイルがサービスの root ディレクトリーに含まれている場合、これらはすべて自動的に生成されます。仕様に関する詳細情報は [API Blueprint Tutorial](#) または [Language Specification](#) を参照してください。

1.4.5. エディター

ビルトインのエディターを使うことで、サービスインスタンスに関連したクラウドコードが編集できます。構文ハイライト表示の機能が組み込まれているため、構文要素を区別しやすくなっています。ユーザーはファイルやフォルダーを新規作成し、保存された変更は自動的に対応する Git リポジトリにプッシュバックされます。

変更がなされたらプレビューが自動的にリフレッシュされるため、変更の反映がすぐに確認できます。'git pull' を実行すると、Git リポジトリから変更がプルされます。

エディターのアプリプレビューでは、多くの選択可能なデバイスで現行アプリがどのように表示されるかを確認できます。

1.4.6. 環境変数

Studio を使用する環境変数は定義が可能で、これらの変数はクラウドコードでアクセスできます。Studio では、各クラウド環境における現在定義済みの環境変数とデプロイ済みの環境変数の一覧を表示します。また、システムが定義する環境変数一覧も見ることができます。

1.4.6.1. アプリの環境変数

このセクションでは、ユーザーが定義した変数が一覧表示されます。これらの変数はクラウドにプッシュされていないことがあるので、値は現在アクティブなものとは異なる場合があります。定義済みの値とデプロイ済みの値で不整合がある場合は、赤字で表示されます。値のフィールドをクリックすると、これを更新することができます。

1.4.6.2. デプロイ済みユーザーの環境変数

このセクションでは、ユーザーが作成したデプロイ済みの環境変数が一覧表示されます。上記と同様に、不整合は赤字で表示されます。ただし、これらの値は直接は更新できません。

1.4.6.3. デプロイ済みのシステム環境変数

このセクションでは、プラットフォームが作成したデプロイ済みの環境変数が一覧表示されます。これらの値はリファレンス目的のみとなり、これら変数の値は変更しないでください。

1.4.6.4. 環境変数の作成

新規の環境変数は、「変数を追加」ボタンをクリックすると作成できます。環境変数の名前は以下のルールにしたがってください。

※ **FH_** または **MONIT_** で始めない。

※ 以下のいずれも使用しない。

■ **PATH**

■ **HOME**

■ **PWD**

- **USER**
- **NODE_PATH**
- **LD_LIBRARY_PATH**

1.4.6.5. 環境変数の設定解除/削除

既存の環境変数一覧で変数にチェックを入れると、変数を設定解除または削除するオプションがあります。

- ※ **設定解除** 選択された環境から変数が削除されます。
- ※ **削除** すべての環境から変数が削除されます。

同時に複数の変数を設定解除または削除することが可能です。

1.4.6.6. 環境変数をプッシュ

変数の値の更新が完了したら、「環境変数をプッシュ」のボタンをクリックして変数をクラウドにプッシュする必要があります。これを実行すると、アプリの再起動も実行されます。

1.4.7. データブラウザー

App Studio データブラウザーセクションでは、開発者は以下のことができます。

- ※ アプリに関連するデータを視覚的かつ対話的に表示する。
- ※ コレクションを表示、作成、削除、インポート、エクスポートする。
- ※ コレクション内のデータを修正する。

詳細情報は [データブラウザーガイド](#) を参照してください。

1.4.7.1. データブラウザーを有効にする

App Studio でデータブラウザータブを選択すると、以下のいずれかの画面が表示されます。

- ※ データブラウザーを有効にしている場合は、データブラウザーが表示されます。
- ※ データブラウザーを有効にすることを要求する画面。
- ※ データを移行してデータブラウザーを有効にすることを要求する画面。

有効ボタンをクリックすると、以下のステップが実行されます。

- ※ アプリが停止されます。
- ※ アプリ用に特別に新規データベースが作成されます。
- ※ mongodb への生接続文字列を含む環境変数がアプリ用に設定されます。

アプリのデータを移行する必要がある場合は、移行ボタンが表示されます。このボタンを押すと、以下が実行されます。

- ※ アプリが停止されます。これで移行中に新たなデータが書き込まれなくなります。
- ※ アプリ用に特別に新規データベースが作成されます。

- ※ 既存のデータが古いデータベースから新しいデータベースに移行されます。
- ※ データが新しいデータベースで検証されます。
- ※ mongodb への生接続文字列を含む環境変数がアプリ用に設定されます。
- ※ すべてが成功すると、古いデータが削除されます。



注記

cloud/application.js および cloud/package.json のコンテンツの更新が必要な場合もあります。必要な場合は移行画面でその旨が知らされます。

データ移行の全ステップが完了したら、アプリの再ステージが要求されます。

1.4.7.2. データブラウザーの使用

1.4.7.2.1. コレクションの表示/追加

Studio のクラウド管理セクションにあるデータブラウザータブを選択すると、アプリに関連付けられているコレクションが表示されます。

この画面には、コレクション一覧上部に以下の 2 つのボタンがあります。

- ※ コレクションの追加
- ※ コレクション一覧のリフレッシュ

コレクション追加のボタンをクリックすると、コレクション名の入力を求められます。作成ボタンをクリックしてコレクションを作成します。

1.4.7.2.2. コレクション内のデータ表示

コレクション内に保存されているデータを表示するには、データブラウザーにあるコレクションのいずれかをクリックします。このビューではこのコレクションに関連付けられたデータが表示されます。画面上部には以下のメインの一覧機能があります。

- ※ コレクションの切り替え。このオプションを選択すると、アプリ用のコレクション一覧が表示されます。表示したいコレクションをクリックしてデータを一覧表示します。
- ※ コレクションにエントリーを追加します。

1.4.7.2.2.1. データの並び替え

特定のフィールドでデータを並び替えるには、一覧上部のフィールド名をクリックします。並び替えは昇順と降順で切り替わります。

1.4.7.2.2.2. データのフィルタリング

表示されたデータをフィルターにかけるには、データブラウザー画面上部にある「フィルター」ボタンを選択します。このボタンをクリックすると、フィルタリングオプションが表示されます。これらのオプションを使用すると、1 つ以上のフィールドで表示されたデータにフィルターをかけることができます。

1.4.7.2.3. データの編集

データブラウザーでは、インラインまたは高度なエディターを使用してデータを編集できます。

- ※ インラインエディターを使用すると、コレクション内の簡単なデータの編集に使用します (例: 単一フィールド内でのテキスト変更)。
- ※ 複雑なデータタイプの編集には高度なエディターが使用されます。これは、対話型の動的エディターや Raw JSON エディターを使用して行われます。

1.4.7.2.3.1. インラインエディターを使用した編集

インラインエディターを使用してエントリーを編集するには、データエントリー右側の編集オプションを選択してから Edit Inline を選択します。オプションが緑色のチェックマークと黒矢印のアイコンに変わります。インラインエディターでの編集が複雑すぎるフィールドの場合、「Advanced Editor Only」のテキストが表示されます。このフィールドは、高度なエディターでのみ編集可能となっています。

エントリーの更新が終わったら、緑色のチェックを選択して変更をデータにコミットするか、黒矢印を選択して変更をキャンセルします。

1.4.7.2.3.2. 高度なエディターを使用した編集

複雑なデータタイプの編集には高度なエディターが使用されます (例: フィールドが複数の入れ子フィールドで構成されている場合など)。

高度なエディターを開くには、データエントリー右側の編集オプションを選択し、Advanced Editor を選びます。高度なエディターには以下の2つのモードがあります。

- ※ フィールドを対話型で追加、編集する動的エディター。
- ※ JSON 形式でデータを直接編集する Raw JSON エディター。

1.4.7.2.3.3. 動的エディターを使用した編集

動的エディターは、JSON データ用の対話型エディターです。各フィールドの構造化ビューを表示し、複雑なデータタイプの編集、追加ができます。アクションメニューでは、エントリーの複雑なフィールドの管理に必要な全機能が提供されます。

- ※ Type: このオプションでは、フィールドのデータタイプをアレイ、JSON オブジェクト、または文字列に変更します。またフィールドを自動に設定することも可能で、その場合、データタイプは入力されたデータから自動的に選択されます。
- ※ Sort: このオプションは、複雑なタイプのサブフィールドを昇順または降順で並び替えます。
- ※ Append: このオプションは、選択したオブジェクトの後にフィールドを追加します。
- ※ Insert: このオプションは、選択したオブジェクトの前にフィールドを追加します。
- ※ Duplicate: このオプションは、選択したオブジェクトをコピーし、それをオブジェクトの最後に追加します。
- ※ Remove: エントリーからフィールドを削除します。

1.4.7.2.3.4. Raw JSON エディターを使用した編集

Raw Editor を使用すると、JSON 構文のデータを編集できます。入力されるデータが有効な JSON フォーマットであることを確認してください。JSON データは、フォーマット化されたフォーム

またはコンパクトなフォームのいずれかで表示できます。

1.4.8. デプロイ

デプロイセクションでは、ユーザーは現行 mBaaS コードのデプロイメントを管理できます。デプロイメントターゲットを作成または選択し、デプロイをライブまたは開発環境にステージすることが可能です。クラウドインスタンスのデプロイの際には、デプロイメントログが表示され、ユーザーはデプロイの詳細を細かく調査することができます。

1.4.9. 統計

Studio では、アプリのエンドポイント統計値とカスタム統計値という 2 種類の統計があります。これらは、クラウドインスタンスの統計セクションにあります。

これらの統計数値を表すグラフには共通の UI 機能があります。データは異なる形式で (例: csv) ダウンロードできます。データの範囲は、各グラフの下にあるスクロールバーとスライダーで変更可能です。グラフ内の各行は、凡例のアイテムをクリックすることでオンとオフが切り替えられます。

1.4.9.1. アプリのエンドポイント

アプリのエンドポイント統計値はすべてのアプリで追加設定なしで使用できます。main.js で定義されているエンドポイントは、エンドポイントが呼び出されると統計値の生成を開始します。「すべてのエンドポイント」の結合グラフも統計値が生成されると利用可能になります。

統計値を生成している各エンドポイントでは、以下のサマリー情報が利用可能となります。

- ※ 1 分あたりのリクエスト数 - 直近 1 分間のリクエスト数 (短い間隔を基に計算された値)
- ※ 平均リクエスト時間 - 直近の間隔でのリクエストあたりの平均時間
- ※ 平均同時リクエスト - 直近の間隔で処理された同時リクエストの平均数 (間隔あたりのリクエスト数と間隔あたりの平均リクエスト時間を基に計算)

デフォルトのグラフビューでは以下のデータが表示されます。

- ※ 1 秒あたりのリクエスト数 - アプリが 1 秒間に処理したリクエスト数 (直近の間隔を基に計算)
- ※ 平均同時リクエスト - 直近の間隔で処理された同時リクエストの平均数 (間隔あたりのリクエスト数と間隔あたりの平均リクエスト時間を基に計算)
- ※ 平均リクエスト時間 (90 パーセンタイル) - 直近の間隔でのリクエストの平均時間。外れ値は無視する。

グラフ上部の凡例にある名前をクリックすると、最長リクエスト時間や最短リクエスト時間といった他のグラフアイテムも表示できます。その下のグラフでは、それらの統計値を含めたものが表示され、1 秒あたりのリクエスト数と平均同時リクエスト数の選択は解除されます。

1.4.9.2. カスタム統計値

カスタム統計値は、[\\$fh.stats](#) API を使って開発者が定義します。

1.4.9.2.1. カウンター

カウンターは線グラフ上で見ることができ、各間隔のカウンター値が表示されます。

**注記**

カウンターは各間隔ごとにリセットされます。

1.4.9.2.2. タイマー

タイマーも線グラフ上で見ることができます。各タイマーの値は、該当間隔におけるそのタイマーの平均値 (90 パーセンタイル) になります。各間隔における上限値と下限値も別の線でグラフ化されます。これらの値により、平均時間においてタイマーがどのように変化したかについての概要が分かります。

1.4.10. 通知およびイベント

アプリでは、アプリに対して特定のアクションが発生すると、イベントが生成されます。これらのイベントはプラットフォームが記録し、App Studio と FHC が開発者に提示します。例えば、開発者がアプリをデプロイすると、プラットフォームはデプロイの発生時間とデプロイ実行者、およびその結果を記録します。

イベントにはカテゴリ、重大度、および名前があります。現在プラットフォームが生成するイベントは以下の通りです。

イベント名	イベントカテゴリ	イベント重大度	説明
CREATE_REQUESTED	APP_STATE	INFO	アプリ作成がリクエストされます。
CREATED	APP_STATE	INFO	アプリが作成されます。
CREATE_FAILED	APP_STATE	ERROR	アプリの作成に失敗しました。
DEPLOY_REQUESTED	APP_STATE	INFO	アプリのデプロイがリクエストされます。
DEPLOYED	APP_STATE	INFO	アプリがデプロイされます。
DEPLOY_FAILED	APP_STATE	ERROR	アプリのデプロイが失敗しました。

イベント名	イベントカテゴリ	イベント重大度	説明
START_REQUESTED	APP_STATE	INFO	アプリの起動がリクエストされます。
START_SUCCESSFUL	APP_STATE	INFO	アプリが起動しました。
START_FAILED	APP_STATE	ERROR	アプリの起動が失敗しました。
SUSPEND_SUCCESSFUL	APP_STATE	INFO	クラウドアプリへの HTTP コールが一定期間 (通常は 1 週間) ないというアクティビティーがない状態になると、クラウドアプリが停止されます。クラウドアプリへの次の HTTP コールで自動的にアプリが再起動します。
STOP_REQUESTED	APP_STATE	INFO	アプリの停止がリクエストされます。
STOP_SUCCESSFUL	APP_STATE	INFO	アプリが停止されます。
STOP_FAILED	APP_STATE	ERROR	アプリの停止が失敗しました。
CRASHED	APP_STATE	ERROR	不明な例外がアプリから送られ、停止されました。システム監視プロセスにより自動的に再起動されます。
KILLED_RESTARTED	APP_STATE	ERROR	アプリが停止し、再起動します。

イベント名	イベントカテゴリ	イベント重大度	説明
TERMINATED	APP_STATE	FATAL	短時間に再起動が過剰に実行されたため (現在は 20 秒間に 10 回の再起動) アプリがシステムにより終了されました。システム監視プロセスが再起動することはありません。
DELETE_REQUESTED	APP_STATE	INFO	アプリの削除がリクエストされます。
DELETED	APP_STATE	INFO	アプリが削除されました。
DELETE_FAILED	APP_STATE	ERROR	アプリの削除に失敗しました。
CHANGE_REQUESTED	APP_ENVIRONMENT	INFO	環境変数の変更がリクエストされます。
CHANGE_SUCCESSFUL	APP_ENVIRONMENT	INFO	環境変数が変更されました。
CHANGE_FAILED	APP_ENVIRONMENT	ERROR	環境変数の変更失敗しました。

1.4.10.1. システムイベントビュー

プラットフォームで生成したイベントはすべて、Studio のクラウドアプリビューにある通知セクションで確認することができます。また、フィルターを使ってイベントを検索することもできます。イベントエントリーをクリックすると、イベントの詳細情報を見ることができます。

1.4.10.2. 警告 & 通知メール

特定のクラウドイベントが発生した際に開発者に通知メールを送信することができます。これは警告で実行されます。警告では、通知メールが実行される条件と使用する E メールアドレスを定義します。

1.4.10.2.1. 警告ビュー

警告ビューには、2つの表があります。1つ目では、該当アプリで作成された全イベントが一覧表示されます。2つ目では、選択された警告設定に合致するクラウドイベントすべてが表示されます。1つ目の表のエントリーをクリックすると、警告の詳細情報が示されます。

1.4.10.2.2. 警告の作成

「警告を作成」ボタンをクリックすると、警告作成ビューが表示されます。このビューでは、警告名、警告が監視するイベント、および警告の送信先となる E メールアドレスを指定します。

イベントを指定する際には、イベントカテゴリー、重大度、イベント名のいずれかまたはその組み合わせを使用します。すべてのフィールドで値を設定する必要はありません。警告が作成され、基準に合致するイベントが発生したら、警告で指定されたアドレスに通知メールが送信されます。

1.4.10.2.3. 通知ビュー

プラットフォームでは送信された通知メールすべての記録を維持し、これは Studio で見ることができます。表内のエントリーをクリックすると、通知の詳細が表示されます。

1.4.11. ログ

ユーザーは、各アプリ環境で作成された現行およびアーカイブ済みのサービスログにアクセスできます。

1.4.12. エンドポイントセキュリティ

エンドポイントセキュリティの機能を使用すると、アプリのエンドポイントに適用するセキュリティレベルを決定できます。

1.4.12.1. エンドポイントセキュリティの管理

エンドポイントセキュリティのオプションは、クラウドインスタンスの左側メニューにあります。セキュリティの設定には、以下の2つの方法があります。

1.4.12.1.1. アプリのセキュリティ

アプリのセキュリティでは、すべてのエンドポイントにわたって適用されるデフォルトのセキュリティレベルを定義します。デフォルトは HTTPS です。アプリ API キーオプションを適用すると、HTTPS は使用されますが、エンドポイントにアクセスするためには、アプリ API キーを送信し、アプリで作成したキーと一致する必要があります。このキーは、アプリの詳細メニューオプションにあります。デフォルトでは、このキーは RHMAP SDK から送信されます。このオプションを有効にすると、エンドポイントは正しい App API キーが送信された場合にのみ、アクセス可能になります。正しいキーがない場合は、401 (Not Authorised) が返され、エンドポイントは呼び出されません。

1.4.12.1.2. 個別のエンドポイントセキュリティ

アプリのエンドポイントに適用するセキュリティは、個別レベルで変更することが可能です。一覧を更新すると、クラウドインスタンスが再起動します。



注記

アプリが実行中でしばらくステージしていない場合は、最新バージョンのソフトウェアを取得するためにアプリを再度ステージする必要があります。

1.4.12.2. 監査ログ

アプリへの変更を透過的に見られるようにするために、セキュリティ設定が変更されると監査ログにエントリーが作成されます。このログは、エンドポイントセキュリティ画面の上部にある監査ログをクリックすると表示されます。監査ログでは、使用中のアプリのセキュリティ設定で実行されたことのログが表示されます。フィルターを使って監査ログを絞り込むこともできます。

1.4.12.3. CORS サポート

CORS (クロスオリジンリソース共有) は、web ページ上の JavaScript がその JavaScript の元になっているドメインではなく、別のドメインに XMLHttpRequests を作成できるようにするメカニズムです。CORS についての詳細は、[CORS on Wikipedia](#) を参照してください。RHMAP では、全クラウドリクエストで自動的に CORS を有効にします。

選択したドメインへのアクセスは、以下のように制限できます。

- ✎ Studio のクラウド管理セクションで「環境変数」をクリックします。
- ✎ **CORS_WHITELIST** という名前の環境変数を追加します。
- ✎ その値を、アクセスを制限するドメインに設定します。
- ✎ 「環境変数をプッシュ」ボタンをクリックして変更を適用します。

第2章 モバイルバックエンド

2.1. MBAAS ターゲットおよび環境

2.1.1. 概要

MBaaS (Mobile Backend as a Service: サービスとしてのモバイルバックエンド) は、クラウドアプリとクラウドサービス向けのランタイムスペースです。各 MBaaS では、Node.js ランタイム、NoSQL データベース、キャッシングサーバー、およびすべての [クラウド API](#) が提供されます。

アプリケーションのライフサイクル管理をサポートするために、**環境** と呼ばれる、クラウドアプリとクラウドサービス向けに提供される分離層があります。環境は、**開発**、**テスト**、および**実稼働**といったプロジェクトの各ライフサイクルステージ向けに設定することができます。環境にはクラウドサービスとフォームが常にデプロイされ、これらはある環境から別の環境に簡単にプロモートすることができます。

1 つの MBaaS では、1 つ以上の環境をホストすることができます。例えば、単一の MBaaS に開発環境とテスト環境を一緒にホストし、別の MBaaS を実稼働環境専用とすることが可能です。複数の環境を MBaaS でホストしている場合でも、それらが共有するのはシステムリソースのみで、Node.js ランタイムやデータベース、キャッシングサーバーなどほとんどのレベルでは分離されており、ホスト名も別のものになります。

2.1.2. MBaaS ターゲットの作成

管理者 > **MBaaS ターゲット** セクションに移動します。

MBaaS ターゲットには、**FeedHenry**、**OpenShift 2** および **OpenShift 3** という 3 つのタイプがあります。各タイプでは、以下に示すように独自の設定が必要になります。



注記

MBaaS ターゲットのタイプは、作成後に変更することはできません。

2.1.2.1. FeedHenry MBaaS ターゲット

✎ **MBaaS ID:** 内部参照用の一意の ID。

このパラメーターは、MBaaS ターゲットの作成後に変更することはできず、特殊文字や大文字を含めることはできません。

✎ **MBaaS ラベル:** Studio 内で使用されるカスタムラベル。

✎ **MBaaS URL:** MBaaS ターゲットのホスト名。

✎ **ユーザー名:** インスタンスにログインする際に使用するユーザー名。

✎ **パスワード:** インスタンスにログインする際に使用するパスワード。

✎ **サービスキー:** FeedHenry MBaaS でシステムの特定に使用するキー。

2.1.2.2. OpenShift 2 MBaaS ターゲット

✎ **MBaaS ID:** 内部参照用の一意の ID。

このパラメーターは、MBaaS ターゲットの作成後に変更することはできず、特殊文字や大文字を含めることはできません。

- ※ **MBaaS ラベル:** Studio 内で使用されるカスタムラベル。
- ※ **MBaaS URL:** パブリック OpenShift インスタンスのホスト名 (例: `openshift.redhat.com`)。
- ※ **ユーザー名:** OpenShift 2 のユーザー名。
- ※ **パスワード:** OpenShift 2 のパスワード。
- ※ **fh-mbaas Host:** 実行中 fh-mbaas サービスのホスト名



注記

アプリでフォームを使用する予定の場合は、**fh-mbaas Host** フィールドの設定が必要になります。これには、fh-mbaas サービスを OpenShift インスタンスにデプロイする追加ステップが必要になります。詳細は、[Enabling Forms support in OpenShift Targets](#) を参照してください。

2.1.2.3. OpenShift 3 MBaaS ターゲット

- ※ **MBaaS ID:** 内部参照用の一意の ID。

このパラメーターは、MBaaS ターゲットの作成後に変更することはできず、特殊文字や大文字を含めることはできません。

- ※ **MBaaS ラベル:** Studio 内で使用されるカスタムラベル。
- ※ **OpenShift Master URL:** OpenShift Master API が利用可能な URL。
- ※ **ユーザー名:** OpenShift 3 のユーザー名。
- ※ **パスワード:** OpenShift 3 のパスワード。
- ※ **OpenShift Router DNS:** OpenShift 3 ルーターのワイルドカード DNS エントリー。例えば、`*.local.feedhenry.io`。
- ※ **Automatic MBaaS Installation:** 自動インストールの MBaaS は、実稼働環境で必要となる回復性と安定性において最適化されていません。実稼働での使用については、[manual installation steps](#) を参照してください。
- ※ **MBaaS サービスキー:** fh-mbaas サービスにおける `FHMBaaS_KEY` 環境変数の値。
- ※ **MBaaS URL:** fh-mbaas が OpenShift で実行している公開ルート

MBaaS ターゲットの設定後には、少なくとも 1 つの環境を追加する **必要があります**。管理者 > 環境 に移動して環境を追加します。

2.1.3. 環境の作成

管理者 > 環境 に移動します。

まず、**環境の作成** ボタンを使用して新規環境を作成します。以下のパラメーターが必要になります。

※ **環境 ID:** 内部参照用の一意の ID。

このパラメーターは、環境の作成後に変更することはできず、特殊文字や大文字を含めることはできません。

※ **環境ラベル:** Studio 内で使用されるカスタムラベル。

※ **MBaaS ターゲット:** 環境が使用する MBaaS ターゲット。1つの環境で使用できるのは1つの MBaaS ターゲットのみです。

これでこの新規環境にクラウドアプリ、クラウドサービス、フォームをデプロイすることができます。

2.2. クラウドリソース

2.2.1. 概要

プラットフォームではアプリがステージまたはデプロイされるとクラウド環境と呼ばれるコンテナ内で実行されることになります。コンテナレベルとアプリレベルの両方における環境リソースの使用率がここで管理、表示できます。

要件

ユーザーは、以下のパーミッションを持つ1つ以上のチームメンバーである必要があります。

※ ドメイン - クラウドリソース (表示 & 編集)

※ ドメイン - プロジェクト (表示)

関連リソース

※ [FHC CLI](#)

※ [Platform API - EndPoint Security API](#)

2.2.2. サマリービュー

ページが読み込まれたら、リソースのサマリービューが表示されます。ドメインで利用可能なクラウド環境や環境の CPU、メモリー、およびディスクの現在の使用率が確認できます。各サマリービューは、クリックすると拡大されます。

2.2.3. リソース

このビューでは、メモリー、CPU およびディスクリソースの詳細な使用率が表示されます。「ダッシュボード」ビューにはこれら3つのリソースすべての概要ビューがあり、各リソースのタブには環境とアプリごとの使用率があります。

2.2.4. アプリ

環境内で現在実行中の全アプリ、それらのステータス、リソース使用率が見られるほか、それらの管理もできます。ここでは組み込みアプリ (ビルド組み込みアプリウィザードでデプロイされたもの) の制御もできることに留意してください。

2.2.5. キャッシュ

環境の現在のキャッシュ使用率が確認できるほか、キャッシュをフラッシュして最大キャッシュサイズをリセットすることができます。

2.3. プッシュ通知

2.3.1. 概要

プッシュ通知は、サーバー側のアプリケーションがモバイルのクライアントとの通信を開始するメカニズムで、サーバー上の新規コンテンツの可用性をユーザーに知らせたり、ショートメッセージを送信するものです。これは通常、メッセージングやソーシャルネットワークアプリケーションに使用されたり、ユーザーがモバイルデバイスを扱っていない場合でも常に情報提供を望んでいる場合などに使用されます。

Red Hat Mobile Application Platform ホスト型 (RHMAP) には、プッシュ通知のサポートが統合されています。単一の統合 API を使用して、全クライアントデバイスに通知を送信することができます。これは単一の関数呼び出しで複数の異なるネットワークにわたって同時に実行することができます。RHMAP のプッシュ通知サポートは、オープンソースのプロジェクトである [AeroGear UnifiedPush Server \(UPS\)](#) をベースにしています。

Studio では、以下の 4 つのセクションでプッシュ通知についての情報が提供されます。

1. Analytics - 詳細は [アナリティクス](#) を参照してください。for more information
2. Variants - 通知の対象となるデバイスグループを作成、管理します。
3. Sender API - Sender API はサポートされており、バックエンドサーバーによるプッシュ通知の送信が可能になっています。
4. Activity log - 全プッシュメッセージの詳細を表示します。

プッシュ通知の使用を開始するには、[client API](#) および [cloud API](#) のドキュメンテーションをご覧ください。いただくか、ステップごとのガイドとなっている [Developing An Application Using Push Notifications](#) を参照してください。

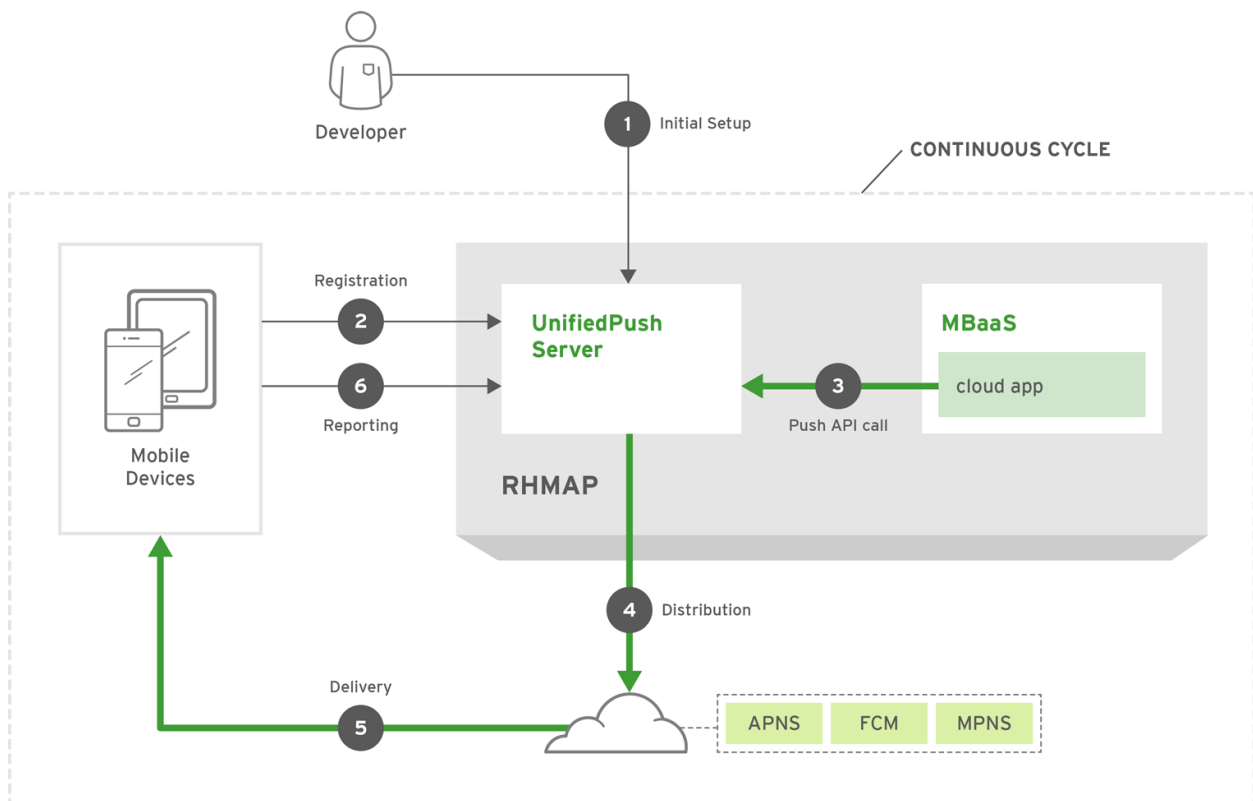
2.3.2. サポートされるプラットフォーム

プッシュ通知は、サポートされている全モバイルプラットフォームで対応するプッシュ通知ネットワークにより利用可能となっています。

- ✳ **Android** — Firebase Cloud Messaging (FCM) (以前は Google Cloud Messaging - GCM)
- ✳ **iOS** — Apple Push Notification Service (APNS)
- ✳ **Windows**
 - ネイティブの Windows Phone 8 アプリと Windows の Cordova アプリの場合は Microsoft Push Notification Service (MPNS)
 - ネイティブの Windows 8.1 および Windows 10 Universal Apps の場合は Windows Push Notification Service (WNS)

2.3.3. 作動方法

RHMAP 内のプッシュ通知サーバーが、クラウドアプリから個別のプッシュネットワークに通知を配布するブローカーとして機能します。個別のネットワークは、登録されたモバイルデバイスにメッセージを配布します。



FEEDHENRY_410718_0716

1. 初期設定

- ※ 各クライアントアプリごとにプッシュ通知のサポートを有効にします。
- ※ FCM や APNS、MPNS など、プッシュ通知ネットワークのすべてのターゲットとなる **バリエント** を有効し、証明書やプライベートキーなどのネットワークへのアクセスのための認証情報を提供します。個別ネットワークの設定については、[Obtaining credentials for push networks](#) を参照してください。Cordova アプリの場合は、ネイティブのクライアントアプリのように、各ターゲットプラットフォーム向けにバリエントを作成します。

2. 登録

起動時にクライアントアプリは **\$fh.push** を呼び出してプッシュサーバーと登録し、通知を受信する準備が整います。登録中は、クライアントの ID と設定がサーバーに送信されます。

3. API コールのプッシュ

クラウドアプリは、**\$fh.push** を呼び出してプッシュ通知を送信します。API ではすべてをターゲットとするか、またはプロジェクト内の特定のクライアントアプリのみをターゲットとすることができます。

4. 配布

\$fh.push の呼び出しは、統合プッシュサーバーが処理し、これに対応するすべてのプッシュネットワークに通知を配布します。ただし、ターゲットとなるクライアントアプリで有効になっているバリエントに依存します。

5. 配信

モバイルデバイスへの実際の配信は、個別のプッシュ通知をネットワークが処理します。

6. レポート

メタデータは RHMAP にレポートバックされ、これには通知が配信されたことおよびエンドユーザーが通知を開いたかどうかについての情報が含まれます。

注記

プッシュ通知がサードパーティーのネットワークを経由することで、これを使用するアプリケーションの設計に以下のような影響が出てきます。

- ※ **配信:** RHMAP ではターゲットのモバイルデバイスへの配信は保証できません。ほとんどのプッシュ通知は配信が保証されないためです。
- ※ **セキュリティ:** プッシュ通知の一部として慎重に扱うべき情報や機密情報は絶対に送信しないことが推奨されます。詳細は、[セキュリティ面の留意点](#) を参照してください。

2.3.4. 通知の送信

一番単純なユースケースは、全体通知を全デバイスに配信する場合です。以下に例を示します。

```
$fh.push(
  {alert: "Hello from {ProductShortName} to all devices!"},
  {broadcast: true},
  function (err, res) {
    if (err) {
      console.log(err.toString());
    } else {
      console.log("status : " + res.status);
    }
  }
);
```

プラットフォームでは、全デバイス、フィルタリングしたデバイスセット、または単一デバイスなどのいくつかのモードによる配信が可能です。また、プロジェクト内のどのクライアントアプリがプッシュ通知を受信するか選択することができます。

2.3.4.1. クライアントアプリのフィルタリング

クラウド側の `$fh.push` 呼び出しでは、以下のいずれかの通知配信モードを明示的に選択する必要があります。

- ※ **broadcast** - 通知はプロジェクト内の全クライアントアプリに配信されます。
- ※ **selected client apps** - App ID で特定される選択したクライアントアプリのみが通知を受信します。

例と完全なドキュメントについては、[cloud Push API documentation](#) を参照してください。

2.3.4.2. 受信者のフィルタリング

クライアントアプリをサーバーに登録する際には、その ID と設定が報告され、通知のフィルタリングを有効にします。プッシュ通知は登録した全デバイスに配信するか、以下の利用可能な基準でフィルタリングを行った特定のサブセットのみに配信するように選択できます。

- ✳ **エイリアス** - ユーザー名やE メール (複数デバイスの場合あり) などのユーザー ID で表されるユーザー。
- ✳ **カテゴリー** - 追加のフィルタリング基準で、**publish-subscribe** 通信モデルを有効にします。
- ✳ **デバイスタイプ** - クライアントデバイスが自動的に設定します。モバイルデバイスハードウェアの短い識別子です。
- ✳ **バリエーション ID** - プラットフォームが自動的に設定します。選択したプッシュ通知の処理にのみ使用可能です。

例と完全なドキュメントについては、[client Push API documentation](#) を参照してください。

2.3.5. セキュリティー面の留意点

2.3.5.1. 通知のコンテンツ

上記で説明したように、プッシュ通知のペイロードは、Google や Apple などのサードパーティーのプッシュ通知ネットワークプロバイダーに配布されます。個人に属する慎重に扱うべき個人または機密情報 (社会保障番号、金融機関の口座もしくは取引情報など)、または個人が安全な通信を合理的に期待する情報は、プッシュ通知の一部として送信しないことが強く推奨されます。

慎重に扱うべきコンテンツをデバイスに配信する必要がある場合の推奨される方法は、サーバーで新たなコンテンツが利用可能であることをユーザーに知らせる通知のみを送信することです。そのようなコンテンツは、ユーザーが通知に反応した場合に別の信頼できるチャンネルで安全にデバイスに送信することができます。

2.3.5.2. バリエーションシークレット

バリエーション ID と **バリエーションシークレット** を格納している設定ファイルは、クライアントアプリバイナリーでモバイルデバイスに配布されます。これらの認証情報は、クライアントアプリがプッシュサーバーに登録する権限を付与します。これらのキーは安全に保管し、クライアントアプリバイナリー以外のチャンネルでは配信しないようにしてください。

バリエーションシークレットキーの安全が侵害された場合は、これを無効にし、新たなキーで置換できます。Studio ではクライアントアプリのプッシュセクションからバリエーションタブを開き、**Renew Variant Secret** をクリックすると、キーが更新されます。

警告

バリエーションシークレットキーを更新したら、既存のインストールはプッシュ通知を使用することができなくなり、新バージョンのクライアントアプリバイナリーで新しいキーを再配布する必要があることに留意してください。

2.3.6. プッシュネットワーク用認証情報の取得

ビルトインの UnifiedPush Server (UPS) が Google、Apple および Microsoft などの個別プロバイダーのプッシュ通知ネットワークにアクセスできるようになるには、承認のための認証情報をプロバイダーのプッシュサーバーに提供する必要があります。必要な認証情報とそれらを取得する方法は、各プロバイダーによって異なります。ターゲットとなるプラットフォームのガイドにしたがい、必要な認証情報を書き留めてください。これは、アプリ用にプッシュネットワークバリエーションを設定する際に RHMAP に提供する必要があります。

2.3.6.1. Android

ステップごとの指示については、[Obtaining FCM Credentials Guide](#) を参照してください。

必要な認証情報:

- ※ サーバー API キー
- ※ プロジェクト番号

2.3.6.2. iOS

ステップごとの指示については、[APNs Push Notifications with AeroGear's UnifiedPush Server Guide](#) を参照してください。

必要な認証情報:

- ※ クライアント SSL 証明書
- ※ 証明書のパスフレーズ

ガイドでは、**バンドル ID** を提供することが求められます。これは iOS アプリの一意の識別子です。iOS アプリを Xcode で作成して RHMAP にインポートしている場合は、既にバンドルIDが設定されている可能性が高くなります。バンドル ID と Xcode でこの設定については、公式 Apple ドキュメンテーションにある [About Bundle IDs](#) を参照してください。

iOS アプリを RHMAP で作成している場合は、アプリにデフォルトのバンドル ID が割り当てられており、これをカスタム値に変更する必要があります。バンドル ID はプロジェクトのプロパティ一覧ファイルで設定されており、**<name-of-your-project>/<name-of-your-project>-Info.plist** の **CFBundleIdentifier** キーの下にあります。

Cordova アプリの場合は、バンドル ID は手動で設定できます。エディターでアプリの config.xml ファイルを開いて、**<widget id="">** フィールドの値を設定します。

警告

生成されたプッシュ通知用のクライアント SSL 証明書は、特定のバンドル ID にバインドされています。このため、バインド ID を変更すると、証明書の再生成が必要となります。

2.3.6.3. Windows

MPNS

設定は不要です。

ステップごとに指示については、[WNS Push Notifications with AeroGear's UnifiedPush Server Guide](#) を参照してください。

必要な認証情報:

- ※ パッケージ SID
- ※ クライアントシークレット

2.3.7. アナリティクス

クライアントアプリに送信されるプッシュ通知についての統計値は、Studio で見ることができます。クライアントアプリの **プッシュ セクション** から **アナリティクス** タブに移動します。以下の統計値が参照できます。

- ✳ 登録デバイスに配信する統合プッシュサーバーに送信されたプッシュメッセージの数
- ✳ プッシュネットワーク経由で登録デバイスに配信された通知の数
- ✳ ユーザーが開いた配信済み通知の数
- ✳ 平均オープン率
- ✳ 現在登録されているデバイス数

プッシュアナリティクス画面では、以下の表も提示されます。

- ✳ オープン率
- ✳ プッシュネットワークごとの内訳

2.3.8. プッシュ通知対応の Cordova アプリの構築

Cordova アプリ内におけるプッシュ通知のサポートは、**aerogear-cordova-push** プラグインで提供されます。RHMAP Build Farm でプッシュ対応のアプリがビルドされる場合は、**www/config.json** ファイルで設定されていれば、このプラグインが特定のビルドに自動的に追加されます。RHMAP Cordova のすべてのテンプレートでは、このプラグインが設定されています。

アプリをローカルでビルドする場合は、以下のコマンドを使用して手動でプラグインを追加する必要があります。

```
cordova plugin add aerogear-cordova-push
```

詳細は、[Using Cordova Plugins](#) を参照してください。

2.3.9. コミュニティーバージョンの UnifiedPush Server

オープンソースコミュニティバージョンの AeroGear UnifiedPush Server (UPS) はアプリでの使用が引き続き可能で、UPS を OpenShift にデプロイし、クラウドサービスの **AeroGear Community Push Connector** でプラットフォームに接続します。このクラウドサービスは、本目的用にプラットフォームで利用可能となっています。ただし、コミュニティバージョンは評価目的のみで利用可能となっており、公式にはサポートされていないことに注意してください。

第3章 DRAG AND DROP APPS

3.1. フォームビルダーを使用したフォームの作成

概要

フォームビルダーを使用すると、既存のフォームの管理のほか、テンプレートから新規フォームを作成することができます。

3.1.1. フォームの作成

フォームはすべて、テンプレートから作成されます。

フォームを作成する際には、**Blank Form** テンプレートを使用した空のフォームを作成するオプションがあります。他には事前設定されたフォームのテンプレートもあり、これらは異なるシナリオ用でよりカスタマイズしやすいものです。

テンプレートはプレビューですべてのフィールドを確認することができます。テンプレートを選択したら、フォーム名とオプションで説明を入力することができます。



注記

フォームのプレビューは、説明目的のみとなります。フォームを提出したり、ドラフトを保存するには、新たなフォームプロジェクトを作成するか、フォームを既存のプロジェクトに関連付ける必要があります。

フォームを作成 ボタンを使ってフォームを作成したら、**フォームを編集** 画面が開いてフォームをカスタマイズできます。

3.1.2. フィールドをフォームに追加する

フィールドを含めることでフォームが作成されます。追加できるフィールドは様々なものがあり、それぞれに別の目的があります。以下ではそれらのフィールドを一覧表示しており、簡単な説明も加えています。フィールドには異なる設定があります。選択したフィールドによっては、別のフィールド設定が利用可能です。

フィールドは以下の2つのカテゴリーに分けられます。

- ※ **コンテンツフィールド** — ユーザー入力 (テキスト、数字、ファイルなど) を受け付けます。
- ※ **構造フィールド** — フォームレイアウトの作成に使用される補助的フィールド。

フィールドは以下の方法でフォームに追加されます。

- ※ **ドラッグ & ドロップ** — フィールドタイプをフォームの好きな場所にドラッグします。フィールドをドラッグすると、フィールドが設置されることになる場所を示す点線の輪郭が表示されます。
- ※ **フィールドタイプをクリックする** — フィールドを1回クリックすると、既に選択しているフィールドの後にそのフィールドが追加されます。

フィールドはフォームに追加された後、フォーム内で好きな場所にドラッグすると再度位置を変えることができます。

3.1.2.1. 構造フィールド

3.1.2.1.1. Section Break

Section Break はフィールドをセクションにグループ化するために使用されます。デフォルトのテンプレートでは、セクションは折りたたみ可能なフィールドのグループです。セクションには、名前と説明を付けてフォームユーザーにセクションが特定できるようにします。

すべてのページに section break が必要なわけではありません。section break のあるページと section break のないページでフォームを作成することは可能です。



注記

ページに section break があり、page break と section break の間にフィールドがある場合は、これらのフィールドは自動的にセクションに組み入れられます。

3.1.2.1.2. Page Break

Page Break が示すのは、フォームの新しいページの始まりです。page break の後のフィールドは、フォームの次のページに含まれます。セクションの場合と同様に、ページに名前と説明を付けてフォームユーザーがこのページを特定できるようにすることができます。

3.1.2.2. コンテンツフィールド

コンテンツフィールドとは、ユーザーがデータを入力するフィールドです。

以下で示すように、フィールドにはタイプ固有の設定を含めることができます。ただし、全フィールドに共通する設定も数多くあります。

- ✦ **Required** — 必須フィールドはフォームの完了時に提出が義務付けられます。
- ✦ **Validate Immediately** — このオプションを使用すると、フィールドに入力された値が入力時に検証されるようになります。これ以外の場合は、ユーザーがフォームの最後のページで **提出** をクリックした際に、フィールドが検証されます。
- ✦ **Admin Only** — このフィールドは Studio の提出エディターでのみ見ることができます。クライアントアプリのフォームには表示されません。また、このフィールドはフィールドルールやページルールの対象とすることはできません。
- ✦ **Repeating** — このフィールドには、**Min** と **Max** の間となる設定可能なエントリー数が含まれます。このフィールドは、フィールドルールやページルールの対象とすることはできません。
- ✦ **Field Code** — フィールドには、フィールドを特定するユーザー定義のコードを割り当てることができます。これは、フォーム内で一意のものとする必要があります。これは、(CSV ファイルのような) 外部のソースからフィールドにデータを追加する際に便利なものです。

3.1.2.2.1. テキストフィールド

テキストフィールドでは、ユーザーは単一行のテキストを入力することができます。文字、数字、特殊文字などが有効な入力となります。

テキストフィールドには、以下の 2 つの検証オプションがあります。

- ✦ **長さ制限** — フィールド内のテキストの **最小** または **最大** 文字数。空白の場合は、テキストの長さに関する検証は実行されません。

- ※ フィールド形式 — テキストフィールドの形式。c で表される文字の組み合わせおよび n で表される文字数、または、フィールドに一致する正規表現で定義されます。

3.1.2.2.2. 段落フィールド

テキストフィールドと同様に、段落フィールドも文字、数字、および特殊文字などすべてのタイプの入力を受け付けます。また、段落フィールドは複数行に及びます。

段落フィールドの検証オプションは以下の 1 つです。

- ※ 長さ制限 — フィールド内のテキストの **最小** または **最大** 文字数。空白の場合は、テキストの長さに関する検証は実行されません。

3.1.2.2.3. 数字フィールド

数字フィールドは単一行のフィールドで、数値の入力ができます。数字フィールドに入力できるのは、数字のみです。文字や特殊文字は受け付けません。

数字フィールドの検証オプションは以下の 1 つです。

- ※ 最小/最大値 — フィールドに入力可能な数字の範囲。空白の場合は、数字入力のサイズに関する検証は実行されません。

3.1.2.2.4. E メールフィールド

E メールフィールドでは、E メールアドレスを入力します。@ と . の両方の記号が必要になります。また、ドメインに少なくとも 1 つの文字入力する必要があります。例えば、**test@example.com** は有効な E メールアドレスになります。

3.1.2.2.5. ウェブサイトフィールド

ウェブサイトフィールドには、ウェブサイトの URL を入力します。

3.1.2.2.6. ドロップダウンメニュー

ドロップダウンでは、ユーザーはドロップダウンのオプション一覧からいずれかのオプションを選択することができます。

オプション自体は、フィールド設定タブの **オプション** セクションに入力すると追加できます。事前選択の値は、追加オプションの横にあるチェックボックスを選択すると、その値を追加できます。

3.1.2.2.7. ラジオボタン

ラジオボタンのフィールドでは、ユーザーはラジオボタン一覧からいずれかのオプションを選択できます。

オプション自体は、フィールド設定タブの **オプション** セクションに入力すると追加できます。事前選択の値は、追加オプションの横にあるチェックボックスを選択すると、その値を追加できます。

3.1.2.2.8. チェックボックス

チェックボックスのコンポーネントでは、ユーザーはオプション一覧からいずれかまたはすべてを選択するか、すべてを選択しないこともできます。

オプション自体は、フィールド設定タブのオプション セクションに入力すると追加できます。事前選択の値は、追加オプションの横にあるチェックボックスを選択すると、その値を追加できます。

チェックボックスフィールドの検証オプションは以下の 1 つです。

- ※ 選択オプションの制限 — 提出可能なオプションの **最小** および **最大** 数。

3.1.2.2.9. マップ

マップフィールドでは、Google Maps API を使ってユーザーの現在地を計算および表示できます。マップ上のユーザーの現在地は、マーカーで示されます。

3.1.2.2.10. 場所

このフィールドでは、マップフィールドで表示されるユーザー位置の座標を表示します。座標は、経度/緯度または偏東/偏北距離で表示することができます。

3.1.2.2.11. ファイル

ファイルフィールドでは提出するファイルを添付することができます。

このフィールドの検証オプションは以下の 1 つです。

- ※ 最大ファイルサイズ — このフィールドを使用してアップロードできるファイルの最大サイズ (単位: キロバイト)。値が入力されない場合は、ファイルサイズは検証されません。

3.1.2.2.12. フォトキャプチャー

画像のアップロードができます。写真を撮ってアップロードするか、ライブラリーから選択する 2 つのオプションがあります。

フォトキャプチャーフィールドを定義する際には、以下のオプションが利用できます。

- ※ Max Height — 撮った写真の最大の高さ (ピクセル単位)。
- ※ Max Width — 撮った写真の最大の幅 (ピクセル単位)。
- ※ Quality — 写真の画質。注: 最近のスマートフォンでは高解像度の写真を撮ることができます。この場合、提出される写真のファイルサイズは大きくなります。
- ※ Photo Source — 写真のソースをカメラ、ライブラリー、またはそのどちらかに制限することができます。
- ※ Photo Type — 写真は JPEG または PNG ファイルで撮ることができます。
- ※ Save To Photo Album — カメラで撮った写真をフォトライブラリーに保存し、使用できるようにします。

3.1.2.2.13. 署名キャプチャー

このフィールドでは、ユーザーの署名をキャプチャーします。

3.1.2.2.14. Datestamp

日付と時間を入力できます。

Datestamp フィールドは、以下の 3 つの方法で定義できます。

- ※ 日付および時間 — ユーザーは日付と時間を入力できます。入力形式は変更可能です。デフォルトは **YYYY-MM-DD HH:mm** です。
- ※ 日付のみ — ユーザーは日付の値のみ入力できます。
- ※ 時間のみ — ユーザーは時間の値のみ入力できます。

3.2. フォームのプロジェクトの管理

3.2.1. 概要

フォームのプロジェクト 画面では、フォームまたはテーマと関連付けられているプロジェクトを表示したり、新規のフォームプロジェクトを作成することができます。

各プロジェクトに関連付けられているフォーム数に加えて、プロジェクトは名前で検索できます。

フォームプロジェクトは、以下のいずれかの方法で作成します。

- ※ プロジェクトを最初から作成する。
- ※ 既存プロジェクトにフォームを追加する。

詳細は、[How to create a Forms Project](#) を参照してください。

要件

フォームプロジェクトを管理するためにユーザーは、以下のパーミッションを持つ 1 つ以上のチームメンバーである必要があります。

- ※ ドメイン — Drag & Drop Apps (表示 & 編集)
- ※ ドメイン — プロジェクト (表示 & 編集)

3.2.2. フォームプロジェクトの作成

新規フォームプロジェクトは、以下のいずれかの方法で作成します。

3.2.2.1. 新規プロジェクトの作成

新規フォームプロジェクトを作成するには **Create New Project** ボタンをクリックし、新規プロジェクト名を入力して **Next** ボタンを選択します。新規プロジェクトには単一のフォームクライアントとクラウドアプリが含まれこれらは関連するフォームと即座に対話します。新規フォームプロジェクトが作成されると、プロジェクトをフォームとテーマで設定するよう指示されます。

3.2.2.2. 既存のプロジェクトに関連付ける

フォームは、Studio ないのプロジェクトであればいずれかのものでも関連付けられます。フォームを既存のプロジェクトに関連付けるには、**Use Existing Project** をクリックしてからフォームに関連付けるプロジェクトを選択し、**Next** をクリックします。

既存のプロジェクトが選択されると、プロジェクトをフォームとテーマで設定するよう指示されます。

3.2.3. プロジェクトにフォームとテーマを追加する

プロジェクトのフォーム & テーマ 画面では、プロジェクトにフォームとテーマを関連付けることができます。

テーマはドロップダウン一覧から選択することで関連付けることができます。テーマをプレビュー表示にするには、プロジェクトテーマ一覧からテーマを選択し、フォーム一覧からいずれかのフォームをクリックします。これでフォームがプレビューエリアに表示されます。

フォームをプロジェクトに追加するには、フォーム一覧をクリックします。これで利用可能なフォーム一覧が表示されます。フォーム名の一部を入力すると、一覧にフィルタリングが実行されます。希望するフォームが見つかったら、一覧から選択します。これでプロジェクトに関連付けられたフォームに表示されます。

フォームの追加が完了したら、**保存** ボタンをクリックしてプロジェクトを更新します。

3.2.4. フォームプロジェクトの設定

高度な設定 画面では、クライアントアプリとクラウドアプリの設定ができます。



注記

これらの設定を更新すると、このプロジェクトのクライアントアプリとクラウドアプリがライブで更新されます。

3.2.4.1. クライアントオプション

クライアントオプションはフォームクライアントアプリで利用可能で、クライアント上のフォームの操作に影響します。

これらの設定はクライアントアプリにデフォルト値を提供しますが、アプリユーザーは下記で説明される **管理者ユーザー** セクションにデバイス ID を追加することでこれらの設定変更ができます。



注記

フォームのテンプレートアプリでは、**設定** セクションに **デバイス ID を表示** ボタンがあり、これで現在のデバイス ID がアプリユーザーに表示されます。

3.2.4.1.1. カメラの設定

カメラの設定は、**フォトキャプチャー** フィールドの操作に影響を与えます。このセクションでは、撮る写真の高さ、幅、および画質を編集することができます。



注記

これらの設定は、**フォトキャプチャー** フィールドの **height**、**width** または **quality** 設定で上書きされます。

3.2.4.2. 提出の設定

このセクションでは、アプリがフォームのデータを提出する際のパラメーターを定義します。

- ※ Max Retries – クライアントアプリがユーザーに提出エラーを知らせるまでの最大提出試行回数。
- ※ Timeout – 提出アップロードがエラーをレポートするまでの秒単位の時間。
- ※ Min Sent Items To Save – デバイス上で保存しておく送信フォームアイテム数を選択する際の、クライアントユーザーのオプション一覧

3.2.4.3. クライアントロギング

このセクションでは、クライアントロギングのパラメーターを定義できます。これらの設定は、フォームの提出で問題がある際に便利なものです。

- ※ Logging Enabled – チェックボックスが選択されていると、クライアントアプリすべてがログをデバイス上に保存します。
- ※ Logging level – クライアントロギングの詳細レベルを設定できます。**debug** ではほとんどがログ記録され、**error** だとクライアント上のエラーのみが記録されます。
- ※ Log Limit – ログが回転するまでにクライアント上で許可されるログエントリーの最大数。
- ※ Log Email – 必要な場合にクライアントがログを送信する E メールアドレス。

3.2.4.4. 管理者ユーザー

管理者ユーザーはクライアントデバイス ID を持っており、デバイス上の設定を変更できます。

デバイス ID を一覧に追加するには、テキストボックスをクリックしてデバイス ID を入力します。



注記

フォームのテンプレートアプリでは、**設定** セクションに **デバイス ID を表示** ボタンがあり、これで現在のデバイス ID がアプリユーザーに表示されます。

3.2.4.5. クラウドロギング

クラウドロギングを有効にすると、このプロジェクトに関連付けられているクラウドアプリすべてのロギングが有効になります。これはフォームの提出で問題がある際に便利なものです。

3.3. フォームフィールドおよびページを動的に表示する

3.3.1. 概要

フォーム内のフィールドは、他のフィールドの値に基づいて動的に表示、非表示とすることができます。

ページも事前定義のルールに基づいて表示、非表示とすることができます。ルールをページに適用する際は、ルールで影響を受ける「オブジェクト」は常にページになります。つまり、ルールの対象はフィールドとなるものの、ページは常に影響されます。ルールの適用はフィールドの場合と同様になります。

3.3.2. ルールの定義

例えば、年齢フィールドにルールを適用する場合、入力が 17 歳以上ならばこれまでの仕事を記載するテキストフィールドが表示されるようにします。年齢が 17 歳未満の場合、仕事のフィールドは表示されないようにします。

- ✧ Subject — ルールが適用されるフィールド。
- ✧ Comparison — Subject の値と特定の値を比較する方法を決定します。比較方法には、**contains** (次の値を含む)、**does not contain** (次の値を含まない)、**begins with** (次の値で始まる)、**ends with** (次の値で終わる) などがあります。
- ✧ Value — ルールが試される値。
- ✧ Action — 条件が満たされた場合に実行されるアクション。表示または非表示のどちらかできます。
- ✧ Object — ルールの条件が満たされた場合に表示または非表示となるフィールドもしくはページ。

ルールを構築する際は、ルールが適用されるフィールドのタイプによっては、異なる比較方法を選択することが可能です。例えば、テキストフィールドと数値フィールドでは異なる比較方法のオプションにすることができます。テキストフィールドにルールを適用する場合は、入力が特定文字で始まる、特定文字を含む、または特定文字で終わるかどうかに基づいてルールを構築することができます。一方、数値フィールドにルールを適用する場合は、特定フィールドへの入力値が特定の値より大きい、小さい、同じかどうかに基づいてルールを構築することができます。

フィールドのタイプによって、選択可能なルール比較方法は異なります。

- ✧ textfield — である、でない、次を含む、次を含まない、次で始まる、次で終わる
- ✧ paragraph — である、でない、次を含む、次を含まない、次で始まる、次で終わる
- ✧ number — より大、より小、等しい
- ✧ email — である、でない、次を含む、次を含まない、次で始まる、次で終わる
- ✧ web — である、でない、次を含む、次を含まない、次で始まる、次で終わる
- ✧ dropdown — である、でない、次を含む、次を含まない、次で始まる、次で終わる
- ✧ radio buttons — である、でない、次を含む、次を含まない、次で始まる、次で終わる
- ✧ checkbox — である、でない
- ✧ date stamp — である、より前、より後



注記

マップ、フォトキャプチャー、署名キャプチャー、ファイルアップロード、および場所の各フィールドにはルールを適用することができませんが、これらは他のフィールドへの入力に基づいて表示または非表示とすることができます。

3.4. データソースの使用

3.4.1. 概要

フォームアプリケーションをより動的なものにするために、フォームの特定のフィールドタイプで値を動的に読み込むように設定することができます。値はデータベースや外部のウェブサービス、バックエンドシステム、および MBaaS サービスでアクセス可能な他のソースから読み込むことが可能です。

データベースは、フォームフィールドと値を提供する MBaaS サービスとを結び付けるメカニズムです。

以下のフィールドタイプではデータソースから値を読み込むことができます。

- ※ ドロップダウン
- ※ チェックボックス
- ※ ラジオボタン
- ※ 読み取り専用

データソースの定義および使用に関するステップガイドは、[Dynamically Populating Form Fields From an MBaaS Service](#) を参照してください。

3.4.2. 作動方法

データソースは、フォームフィールドと MBaaS サービスのエンドポイントを結び付ける役割を果たします。フォームの定義にアクセスがあった場合、データソースで支えられているフィールドのデータは、直接 MBaaS サービスから取得されません。データソースは常に環境でキャッシュされているデータのバージョンを返します。

例えば、**開発** 環境にデプロイされているフォームからデータソースが呼び出された場合、関連する MBaaS サービスも **開発** 環境にデプロイされている必要があります、かつデータはこの**開発** 環境向けにのみキャッシュされます。

3.4.2.1. リフレッシュ期間

MBaaS サービスからキャッシュされたデータは、設定期間にしがってクラウド側で定期的によりフレッシュされます。この期間は 1 分間から 7 日間の間とすることができます。

データソースのリフレッシュがなんらかの利用で失敗すると、バックオフアルゴリズムが徐々にリフレッシュ期間を拡大し、最大 7 日間とします。この期間中にデータソースの更新が自動または手動更新で成功すると、データソースは元の設定された期間に戻ります。

3.4.3. 要件

- ※ データソースとして使用される MBaaS サービスは、**fh-mbaas-api** のバージョン **5.8.1** 以上を使用する必要があります。
- ※ データソースを使用するフォームに関連付けられているクラウドアプリは、**fh-mbaas-api** のバージョン **5.0.0** 以上を使用する必要があります。

3.4.3.1. JSON 形式

データソースとして使用される MBaaS サービスの HTTP エンドポイントは、以下の形式の JSON 応答を返す必要があります。

- ※ ドロップダウン、チェックボックス、およびラジオボタンのフィールドタイプの場合:

以下の 3 つのキーを伴うオブジェクト一覧:

- **key** – (オプション) 内部識別子で、HTML フォームフィールドで **name** 属性として使用されます。
- **value** – ユーザーフレンドリーなテキスト表示
- **selected** – (オプション) ラジオボタンおよびドロップダウン の場合の単一オプション、または チェックボックス の場合の複数オプションの事前選択に使用します。

```
[
  {
    "value": "January",
    "selected": "true"
  },
  {
    "key": "02",
    "value": "February"
  },
  {
    "value": "March"
  },
  ...
]
```

注記

サービスが返すオブジェクト数には制限がありません。また、返されたオブジェクトはすべてオプションとして表示されます。オプションを多く使い過ぎるとフォームの移動が難しくなります。

※ **読み取り専用** テキストフィールドの場合:

```
[
  {
    "value": "<b>Note:</b> You can use <i>HTML</i> in the Read Only field."
  }
]
```

HTML のサブセットは **value** でサポートされており、フォームに表示されます。

一覧に複数のオブジェクトがある場合は、全オブジェクトからの **value** フィールドの値は改行で分けられ、連結されます。

3.4.3.2. MBaaS サービスエンドポイント

MBaaS サービスは、**fh-mbaas-api** の必須バージョンを使用していて、かつ HTTP エンドポイントの出力が上記の必須形式に一致していれば、いずれのものでもデータソースとして使用できます。

デフォルトでは以下の 3 つの MBaaS テンプレートが提供され、これらは一般的なユースケースを示します。

- ※ 静的データソース – MBaaS サービスの JSON ファイルで静的に定義されたデータを提供します。
- ※ MongoDB データソース – ビルトインの MongoDB データベースに保存されたデータを `$fh.db` に読み込んで提供します。
- ※ インターネットデータソース – 外部のウェブサイトから解析されたデータを提供します。

3.5. テーマ

概要

テーマは、フォームの外観をカスタマイズするために使用されます。テーマはフォームエディターで作成、管理できるほか、フォームの外観をパーソナライズするためにフォームに関連付けることもできます。カスタマイズ可能なテーマの要素には、テーマのロゴや背景色、文字の体裁、罫線の設定などがあります。

3.5.1. テーマの作成

テーマの作成時には、テンプレート一覧からテーマを選択します。

テーマのテンプレートを選択したら、そのテーマがプレビュー表示されます。テンプレートを選択したら、テーマ名および説明 (オプション) を入力します。これでテーマを編集して、フォームの外観をカスタマイズできるようになります。

3.5.2. テーマの編集

3.5.2.1. ロゴ

ロゴは、ロゴセクションのファイルのアップロードオプションでアップロードします。ロゴとして使用する画像は、PNG か JPEG 形式である必要があります。

3.5.2.2. 背景

背景色 セクションでは、テーマの異なる部分で背景色を変更することができます。セクション名の横にある色の付いた四角をクリックし、ピッカーから色を選択することで設定できます。色のピッカーを閉じると、更新されます。

3.5.2.3. 文字の体裁

文字の体裁のセクションでは、フォントサイズとテキスト装飾がカスタマイズできます。選択したフィールドの色も、セクションタイトルの横にある色の付いた四角をクリックすると変更できます。四角をクリックすると、色のピッカーが表示されます。

3.5.2.4. 境界線

テーマの境界線セクションもカスタマイズが可能です。色、罫線のタイプ、罫線の幅に基づいてカスタマイズを実行できます。

3.5.2.5. スペース

いくつかのフィールドは、フィールド間のマージンとパディングのカスタマイズが可能です。



注記

各セクションでの変更でテーマがどのようなようになったかをプレビュー表示するには、プレビューエリアの下部にあるドロップダウンメニューからフォームを選択します。

3.6. 提出の表示

3.6.1. 概要

提出 ページでは、Studio またはクライアントアプリで提出されたフォームを確認できます。提出はフォームごと、プロジェクトごと、または最近の提出ごとに表示できます。

フォームの提出は提出ビューワで表示され、提出エディターで編集できます。

提出は、**Export Submissions** ボタンを使用するとカンマ区切り値 (CSV) の形式でエクスポートできます。提出をエクスポートする際は、提出されるフィールドは以下のヘッダーとすることができます。

✦ **Field Name** — フィールド名

✦ **Field Code** — フィールドに割り当てられるフィールドコード。割り当てられていない場合は、代わりに **Field Name** が使用されます。



注記

Field Names と **Field Codes** は、[Form Builder](#) を使ってフォーム内のフィールドに割り当てられます。

「最近の提出」ページでは、最近提出されたフォーム一覧が表示されます。フォーム名やプロジェクト名以外にも、プロジェクトがデプロイされている現在の環境や提出を受け取った日時、フォームフィールドへの入力など、様々な情報が提供されます。

提供をクリックすると、その詳細が表示されます。

3.6.2. 提出の並び替え

3.6.2.1. フォームごとの並び替え

提出の **フォームごと** のページでは、フォームベースの提出が表示できます。ドロップダウンリストからフォームを選択し、その特定のフォームに関連する個別の提出についての情報を確認できます。

一覧から提出を選択すると、提出の追加情報が表示されます。下にはフォームのプレビューと追加情報が表示されます。

フォームの提出セクションでは、各フィールドに入力された値とともに、提出済みのフォームがプレビュー表示されます。

提出情報セクションでは、提出に関連付けられているアプリ ID、デバイス ID、および IP アドレスなどのより技術的な情報が表示されます。



注記

提出エディターでフォームの提出に変更が加えられると、「Date time edited」と「Edited by」のフィールドに変更が記録されます。

3.6.2.2. プロジェクトごとの並び替え

提出のプロジェクトごとのページでは、プロジェクトベースで提出を表示できます。まず、一覧からプロジェクトを選択し、それからそのプロジェクトに関連するフォーム一覧からフォームを選択します。

3.6.2.3. 提出のフィルタリング

フォームごと とプロジェクトごとの提出ページには、検索フィールドがあります。検索ボタンをクリックすると、以下のフィールドに対して値が比較されます。

- ✎ 提出の ID
- ✎ 提出がなされたフォーム名
- ✎ 提出がなされたフォームの ID
- ✎ 提出したデバイスの ID
- ✎ 提出の発信元となった IP アドレス

3.6.3. 提出の検索

提出の高度な検索では、カスタマイズした基準でフォーム提出を検索することができます。ユーザーは、関連する提出を見つけるために、フィールドの値にフィルターをかける条項を構築することができます。

3.6.3.1. 検索基準の作成

あるフォームの提出を検索するには、以下を実行します。

1. 利用可能なフォームの一覧からフォームを選択します。これで **Single Fields** と **Repeating Fields** のボックスがタイプに一致するフィールド一覧で設定されます。
2. + をクリックして条項を追加するか、- をクリックして条項を削除します。
3. フィールドを選択して条項を設定し、条項のタイプ(である、でない、など)を選択して比較する値を入力します。
4. 全条項が作成されたら、**Search** をクリックして作成した条項に一致する提出一覧を探します。



注記

フィールドが繰り返しの場合は、このフィールドの各エントリーが条項に対して比較されます。



注記

作成した検索基準に一致する提出は、**Export Submissions** ボタンを使用してエクスポートできます。

3.6.4. 提出詳細の表示

ユーザーの提出した管理者フィールド以外のものはすべて、ユーザーがアップロードした画像とファイルとともに表示されます。

また、追加情報を提出に追加するために **Admin Only** フィールドも表示されます。

画像はインラインで表示されます。画像をコンピューターに保存するには、以下を実行します。

1. 画像を右クリックします。
2. **Save Image As** を選択します。
3. 画像を保存する場所を選択します。

各ファイルエントリーのリンクをクリックすると、ファイルが開きます。

ファイルをコンピューターに保存するには、以下を実行します。

1. 画像を右クリックします。
2. **Save Link As...** を選択します。
3. ファイルを保存する場所を選択します。

画面下部のボタンには以下の 4 つの機能があります。

- ※ PDF のダウンロード: 提出を PDF ドキュメントに変換できます。変換が完了すると、このプロセスはブラウザーでダウンロードを開始します。
- ※ 提出の印刷: ドキュメントを印刷可能な形式に変換できます。完了すると、このプロセスはブラウザーで印刷機能を開始します。
- ※ 提出の削除: ユーザーは提出の削除を確認するよう求められます。
- ※ 提出の編集: 提出を編集して追加データを追加できます。

注意

提出を削除すると元に戻すことはできません。ファイルを含む、その提出に関連するすべてのコンテンツが削除されます。

3.6.5. 提出の編集

フィールドにデータを入力すると (例: テキスト、数字)、提出を編集することができます。

管理者フィールドは、**(管理者)** アイコンで示されます。

繰り返しフィールドの場合、追加の値は **Add Input** ボタンでフィールドに追加できます。

ファイルまたは写真フィールドのコンテンツは、ファイルまたは写真の下にある **Choose File** をクリックすると変更できます。ファイルまたは写真は、自分のローカルコンピューターから追加できます。新しいファイルを選択すると、そのファイルがアップロードのキューに登録されたこと示すメッセージが表示されます。

提出への変更を保存するには、**Update Submission** ボタンをクリックします。これにより、提出データが更新され、選択されたファイルがアップロードされます。

Cancel をクリックすると、提出に入力された変更は中止されます。

3.6.6. 全提出で通知を受ける

通知セクションでは、ユーザーは提出が受信された際に受け取る自動 E メールメールアドレスを管理することができます。

ここでは、メールアドレスを通知リストに追加したり、リストから削除することができます。リストに追加されたメールアドレスでは、そのフォームのコピーが提出されると E メールを受け取るようになります。

第4章 運営管理

4.1. アカウント

概要

アカウント設定セクションには多くの機能があります。ユーザーは、アカウントタイプや関連する E メールなどのアカウントの概要情報を確認できます。また、ユーザーは自分のアカウントパスワード設定をここで管理できます。API および SSH キーもアカウントセクションで管理できます。

要件

アカウント情報を確認するための特別な要件はありません。

関連リソース

🔗 [FHC CLI](#)

🔗 [Platform API - API Keys](#)

4.1.1. アカウントの管理

アカウントのこのセクションでは、名前や E メールアドレス、およびアカウントタイプなどの詳細が表示されます。導入紹介を再度有効にしたりコンテキストヘルプの常時表示などのアカウント設定は、ここで修正できます。パスワードもこのページで変更できます。

4.1.2. SSH キーの管理

アプリに Git 経由でアクセスするには、SSH 公開キーをアップロードする必要があります。アカウントのこのセクションでこれを行います。各種マシンで異なるキーを使用している場合、または他の人にアクセスを許可したい場合は、複数のキーをアップロードできます。キーはそれに付けたラベルで検索したり、削除することもできます。

SSH キーは通常、以下の場所に格納されています。

```
~/.ssh/id_rsa.pub
```

新しいキーを生成するには、以下を使用します。

```
ssh-keygen -t rsa -C "your_email@example.com"
```

警告

ここで追加された SSH キーには、プロジェクトタブで表示されるすべてのプロジェクト/アプリの Git レポジトリにプッシュ & プルすることができます。

警告

SSH キーをアップロードできるのは、1 回のみです。既存のキーをアップロードしようとすると、別のアカウントでも失敗します。これは、キーをユーザーのアカウントにマッピングする上で必須の動作です。複数の SSH キーを使用する必要がある場合は、[guide for configuring your local SSH to use multiple keys](#) を参照してください。

4.1.3. API キーの管理

従来のウェブアプリケーションでは、ユーザーがサーバーと認証した後に、そのユーザーに送信される全リクエストにクッキーが組み込まれていました。RHMAP ではこの方法の代わりに API キーを使用します。これはクッキーと類似していますが、**X-FH-AUTH-USER** と呼ばれるヘッダーで送信されます。このヘッダーは、ユーザーに送信されるすべてのリクエストに組み込まれる必要があります。

ユーザーの API キーは、App Studio で **My Account** → **My Profile** → **API キー** に移動すると確認できます。App Studio の **API キー** セクションでは、API キーの作成、一覧表示、編集、取り消しができます。プログラムで REST API にアクセスする際には、有効な API キーを各リクエストの "X-FH-AUTH-USER" ヘッダーに手動で設定する必要があります。

デフォルトでは、ユーザー API キーはありません (**FH_MBAAS_API_key** などのシステムが作成したキーを除く)。API キーは複数の作成が可能なので、それぞれに一意のラベルを付けるとよいでしょう。キーの値は自動的に生成されます。必要な場合は、後でラベルを修正できます。特定のキーが不要になった場合、またはそれをもう使用しないようにする (セキュリティのため) 場合は、単に削除または取り消します。

4.2. 管理

概要

プラットフォームの管理セクションでは、ユーザーや認証ポリシー、デプロイターゲットなどの管理ができます。また、エンドユーザーのデバイスにアプリを配信する上でのすべての面を管理したり、その使用を追跡、管理することができます。

要件

ユーザーは、以下のパーミッションを持つ 1 つ以上のチームメンバーである必要があります。

- ✳️ ドメイン — 認証ポリシー (表示 & 編集)
- ✳️ ドメイン — 認証ポリシー (表示 & 編集)
- ✳️ ドメイン — デプロイターゲット (表示 & 編集)
- ✳️ ドメイン — App Store (表示 & 編集)
- ✳️ ドメイン — 管理者 (表示 & 編集)

関連リソース

- ✳️ [FHC CLI](#)
- ✳️ [Platform API - MAM](#)

- ✳ Platform API - User Administration

- ✳ Platform API - User Roles

4.2.1. ユーザー

管理者は、ユーザー作成、ロールの割り当て、ユーザーの無効化、およびユーザーの削除などのユーザー管理に関わるすべてのことが実行できます。

以下に利用可能な全オプションを記載します。

- ✳ ユーザー詳細の表示、作成、更新。これはアプリユーザー ([\\$fh.auth](#) API との使用) でも App Studio ユーザーの場合でも該当します。
- ✳ ユーザーへのロールの割り当て
- ✳ ユーザーの無効化 - ユーザーがログインできないようにします。
- ✳ データ消去のためにユーザーをマーク ([\\$fh.auth](#) と使用して、このフラグを確認してユーザーのアプリ/デバイスからデータを消去できます)
- ✳ 招待メールをユーザー送信/再送信
- ✳ ユーザーの削除

4.2.1.1. ユーザーの表示

ユーザー一覧エリアでは、ユーザー ID や名前、メールアドレス、ステータスなどの詳細と最終ログイン日時が表示されます。特定のユーザーを検索するオプションもあり、これは入力すると表示が自動的に更新されます。また、新たなユーザーを作成したり、特定のユーザーを編集するオプションもあります。新たなユーザーの詳細は、ユーザー編集エリアで確認できます。

一覧ビューからユーザーを削除することもできます。実際に削除する前に、確認が必要になります。ユーザーの削除が可能なのは、そのユーザーが作成したアプリが Studio がない場合のみです。アプリを所有するユーザーを削除しようとすると、エラーが返されます。

4.2.1.2. ユーザーの作成

新規ユーザーを作成するには、ユーザー ID を指定する必要があります。ユーザー ID はユーザーの一意的識別子で、ドメイン内で一意である必要があります。変更はできません。既存のユーザー ID と重なる場合は、エラーメッセージが出ます。オプションでメールアドレス、名前、パスワードも指定することができます。ユーザーに招待メールを送信する場合は、そのユーザーのメールアドレスを指定する必要があります。「招待メールを送信します」にチェックを入れると、そのユーザーは自分のパスワードを指定できるようになります。

新規ユーザーにロールを割り当てることもできます。ユーザーのロールに関する詳細は、下記で説明します。

「認証ポリシー」セクションでは、ユーザーを認証ポリシーに関連付けることができます。

4.2.1.3. ユーザーの更新

名前やメールアドレス、パスワードといったユーザーのフィールドは更新が可能です。ユーザーのパスワードは表示できませんが、異なるものに変更することは可能です。

また、ユーザーがメンバーとなっているチームを表示することも可能です。あるチームのメンバーとしてユーザーを追加するには、チームのボックスをクリックし、追加先となるチームを選

択します。

このビューでは、ユーザーを有効または無効にすることができます。ユーザーを無効にすると、そのユーザーは `$fh.auth` API での認証ができなくなります。API が特別な値を返して、ユーザーが無効になっていることを知らせます。

ここには「ユーザーを一完全に削除」するオプションもあります。これを選択した場合は、以下のことが実行されます。

- ※ ユーザーが無効にされます。
- ※ このユーザーが `$fh.auth` を使用してログインすると、既存の全アプリデータを削除すべきという特別な値を API が返します (データ削除機能の実装は、開発者の判断になります)。

詳細は、`$fh.auth` API を参照してください。

4.2.1.4. 招待メール

招待メールには一意のリンクが含まれており、これを開けるとユーザーはパスワードを入力することでアカウントのアクティベーションができます。このメールのメッセージは通常、以下のようになります。

Hello User,

Your account has been created and is awaiting activation.

To activate your account, please click on the link below where you can also set your password.

[https://\[your-studio-domain\].feedhenry.com/studio/activate.html?t=123456789012345678901234](https://[your-studio-domain].feedhenry.com/studio/activate.html?t=123456789012345678901234)

Regards

新規ユーザーの作成時には、招待メールを自動送信するオプションがあります。別の方法では、ユーザーの更新エリアに招待メールを再送信するオプションもあります。

4.2.2. ユーザーロール

注記

Red Hat Mobile Application Platform ホスト型のパーミッションシステムは、FeedHenry v2 プラットフォームで使用されていたロールベースではなく、チームの概念に基づいています。チームベースのパーミッションシステムがアクティブな場合、Studio 内のユーザープロファイルには **チーム** セクションが含まれます。このシステムがアクティブではない場合は、**ロール** ベースのパーミッションシステムがアクティブとなります。ロールとチームの比較については、[こちら](#) を参照してください。

ユーザーのロールはユーザーアカウントに追加され、これによりプラットフォーム内での異なる権限が付与されます。ユーザーはアカウントに割り当てられたロールによって、異なるタスクを実行したり、プラットフォームの異なるセクションにアクセスできるようになります。例えば、プラットフォームの **アナリティクス** セクションにアクセスできるのは、**アナリティクス** のロールがあるユーザーのみです。アカウントにユーザーのロールを追加したり削除したりする権限があるのは、**Admin** または **Domain Admin** ユーザーのみです。

4.2.2.1. アナリティクス (analytics)

- ※ プラットフォームのアナリティクスセクションにアクセスできる。
- ※ プロジェクトの使用率を監視できる。

4.2.2.2. フォームエディター (formseditor)

- ※ フォームやテーマを作成できる。
- ※ フォームプロジェクトを作成できる。
- ※ フォーム、テーマ、および自分のグループに関連付けられているプロジェクトを編集できる。
- ※ フォームとテーマを該当するグループのプロジェクトに関連付けることができる。

4.2.2.3. フォーム管理者 (formsadmin)

- ※ フォームやテーマを作成できる。
- ※ フォームプロジェクトを作成できる。
- ※ グループを作成できる。
- ※ ユーザーをグループに割り当てられる。
- ※ 特定のドメインで作成されたすべてのフォーム & テーマを表示、アクセス、管理できる。
- ※ ドメイン上の全プロジェクトからの提出を表示できる。
- ※ ドメイン上のプロジェクトからの提出を編集できる。



注記

グループにユーザーを追加するのは、フォーム管理者の責任です。フォームエディターはグループに関連付けられておらず、フォームやテーマを編集する権限はあるものの、作成したフォームやテーマを見ることはできません。

4.2.2.4. 提出ビューワ (submissionsviewer)

- ※ グループ内のプロジェクトからの提出を見ることができる。

4.2.2.5. 提出エディター (submissionseditor)

- ※ グループ内のプロジェクトからの提出を見ることができる。
- ※ グループ内のプロジェクトからの提出を編集することができる。

4.2.2.6. 開発者 (dev)

- ※ すべてのタイプのプロジェクトを作成、管理できる。
- ※ 利用可能なプラットフォーム用にアプリバイナリーを構築できる。
- ※ プライベートキーや証明書などの開発リソースをアップロードできる。

- ※ デプロイメントターゲットの作成、編集、およびメンテナンスができる。
- ※ パブリックサービスをプロジェクトに追加できる。

4.2.2.7. 開発管理者 (devadmin)

- ※ すべてのタイプのプロジェクトを作成、管理できる。
- ※ 利用可能なプラットフォーム用にアプリバイナリーを構築できる。
- ※ プライベートキーや証明書などの開発リソースをアップロードできる。
- ※ デプロイメントターゲットの作成、編集、およびメンテナンスができる。
- ※ 特定ドメイン上のプロジェクト & アプリすべてを表示、それらにアクセスできる。
- ※ 特定ドメイン上のデプロイメントターゲットすべてを表示、編集できる。

4.2.2.8. サービス管理者 (serviceadmin)

- ※ mBaaS サービスのプロビジョニングができる。
- ※ mBaaS サービスの管理ができる。

4.2.2.9. ドメイン管理者 (portaladmin)

- ※ 認証ポリシーを作成、編集できる。
- ※ ドメイン向けのアプリストアの更新ができる。
- ※ App Store のアプリを作成、管理できる。
- ※ App Store のグループを作成、管理できる。
- ※ App Store のデバイスを管理できる。
- ※ App Store のログを表示できる。
- ※ 管理者ページの「モバイルアプリの管理」セクションにアクセスできる。
- ※ ユーザーの追加と削除ができる。
- ※ ユーザーにロールを割り当てることができる。

4.2.2.10. 顧客管理者 (customeradmin)

- ※ この顧客に属するドメイン内のユーザーを管理できる。

4.2.2.11. リセラー管理者 (reselleradmin)

- ※ 顧客に属するドメイン内のユーザーを管理できる。
- ※ ユーザーに顧客管理者のロールを割り当てることができる。

4.2.2.12. 管理者 (admin)

- ※ プラットフォーム内ですべてのアクションを実行できる。

- ※ ユーザーにリセラー管理者と顧客管理者のロールを割り当てることができる。

4.2.3. 認証ポリシー

プラットフォームには、豊富な認証および承認のメカニズムがあります。そのコアとなるのは認証ポリシーです。認証ポリシーを使用すると、ユーザーがアプリケーションと App Store のアプリにアクセスする際に認証する方法を定義でき、さらに各種の認証チェックも定義できます。

RHMAP では以下の認証プロバイダーに対応しています。

- ※ OAuth: 明確には OAuth2。これを使用すると、Google のような OAuth プロバイダーに対してユーザーを認証することができます。
- ※ LDAP: Active Directory と Open LDAP サーバーがサポートされており、通常は「エンタープライズ」タイプの統合に使用されます。
- ※ FeedHenry: RHMAP プラットフォームの認証メカニズム。
- ※ MBAAS: [mBaaS Service](#) 経由でユーザーを認証します。つまり、どのような認証メカニズムでも使用可能になります。

4.2.3.1. 認証ポリシーの表示

認証ポリシー一覧エリアでは、ポリシー ID や認証タイプなどの詳細が表示されます。新規ポリシーを作成するオプションや既存ポリシーを編集するオプションがあります。

4.2.3.2. 認証ポリシーの作成

認証ポリシーを作成する際には、ポリシー ID と認証タイプの特定情報を指定する必要があります。

4.2.3.3. OAuth 認証タイプ

OAuth 認証では、ユーザーがあるサービスにアクセスするためにユーザー名やパスワードを知らせることなく、アプリケーション開発者はそのユーザー固有の情報にアクセスできます。基本的な流れは以下のようになります。

- ※ アプリケーションにログインを希望するユーザーが Google オプションでサインインします。
- ※ アプリケーションがリクエストトークンをプロバイダーから取得し、そのトークンを持つユーザーをプロバイダーにリダイレクトします。
- ※ ユーザーが自分のアカウントにログインし、要求された情報へのアクセスを許可し、アプリケーションに再度リダイレクトされます。
- ※ アプリケーションが OAuth トークンと OAuth 検証ツールを受け取り、これがプロバイダーと OAuth アクセストークンを交換します。
- ※ アクセストークンは、ユーザー情報にアクセスを要求する将来のリクエストとともに送信されます。

4.2.3.4. LDAP 認証タイプ

以下の LDAP 認証方法がプラットフォームではサポートされています。

- ※ Simple: 簡易バインド。ユーザー DN とパスワードが LDAP サーバーに送信されます。このオプションは、追加設定なしで、Active Directory と OpenLDAP でサポートされています。
- ※ SASL: Digest MD5 および CRAM MD5 という 2 つのタイプの簡易認証およびセキュリティ層がサポートされています。サポートされている SASL タイプについての情報は、お使いの LDAP サーバーのドキュメンテーションを参照してください。

4.2.3.4.1. URL

URL フィールドは、お使いの LDAP サーバーのアドレスで、例えば `ldap://foo.example.com:389` になります。「389」は標準の LDAP ポートです。

4.2.3.4.2. DN 接頭辞

「識別名 (dn) 接頭辞」フィールドは、完全ユーザー DN の接頭辞です。認証に必要とされるユーザーの完全 DN の形式は、通常以下ようになります。**<prefix><user><dn>**、例:

uid=fred,ou=people,dc=example,dc=com。接頭辞は通常 Active Directory の場合は「cn」、Open LDAP の場合は「uid」になります。この設定については、LDAP 管理者に問い合わせてください。また、これは必須の設定ではなく、空白にしておくことも可能です。

4.2.3.4.3. DN

「識別名 (dn)」フィールドは、ユーザーへの完全ディレクトリーパスの提供に使用されます。上記の例の場合は、**ou=people,dc=example,dc=com** になります。これも空白にしておくことが可能です。この設定に関しては、LDAP 管理者に問い合わせてください。

4.2.3.4.4. DN および DN 接頭辞

DN および DN 接頭辞のフィールドは、プラットフォームが LDAP サーバーへの Bind に使用する名前の構築のために使用されます。**uid=fred,ou=people,dc=example,dc=com** の例では、接頭辞は **uid=** に、**dn** は **,ou=people,dc=example,dc=com** に設定すると、ユーザーは **fred** をユーザー名として使用できます。これらのフィールドは空白にしておくことが可能で、その場合はプラットフォームがユーザー入力 of データを使用してバインドを試みます。これらのフィールドは、**userPrincipalName** など、LDAP サーバーがバインディングに許可する値の構築に使用できます。**userPrincipalName** がメールアドレスの場合は、**dn** フィールドを使用して共通のドメイン名を追加できます。例えば、**dn** フィールドが **@example.com** に設定され、ユーザーがログイン名 **fred** を使用すると、プラットフォームは **fred@example.com** として LDAP サーバーへのバインドを試みます。

4.2.3.5. Active Directory の設定例

以下は、Active Directory (AD) サーバーに対してユーザーを認証する指示の例です。

- ※ まず、サーバーが RHMAP クラウドからアクセス可能となっている必要があります。プライベート RHMAP デプロイの場合、全サービスが同一のファイアウォールの内側にあるため、これは通常問題ではありません。パブリック RHMAP クラウドの場合、AD サーバーにアクセスするために VPN を設定するか、これについて Red Hat と相談する必要があります。または、AD サーバーを公的にアクセス可能とする必要があります。
- ※ 次に認証ポリシーを作成して、以下の値を指定します。
 - デフォルトでは、AD は「簡易」認証メソッドを設定なしでサポートするので、認証メソッドで「簡易」を選択します。

- URL フィールドには、**ldap://foo.example.com:389** などのお使いの AD サーバーのアドレスを入力します。
- 「DN 接頭辞」と「DN」のフィールドは空白にしておきます。これらはデフォルトでは AD で不要です。ユーザーが認証を行うと、メールアドレスのみが AD に渡され、AD ユーザーの特定では通常これで十分です。

4.2.3.6. FeedHenry 認証タイプ

プラットフォーム認証は、ユーザーの認証方法として使用することができます。この方法を使用する場合は、ユーザーはプラットフォームを終了する必要があります。プラットフォームユーザーの運営方法については、ユーザーセクションを参照してください。

FeedHenry 認証ポリシータイプを選択した場合は、特に追加の設定オプションは必要ありません。認証タイプで「FEEDHENRY」を指定するだけです。

4.2.3.7. MBaaS 認証タイプ

このタイプでは、[mBaaS Services](#) からのユーザー認証がサポートされます。コアプラットフォームでサポートされていない認証タイプをアプリケーションが必要とする場合は、これをサポートするために新たな mBaaS サービスを簡単に作成できます。

4.2.3.7.1. サービス

この認証ポリシーが使用する mBaaS サービス名です。最初に mBaaS サービスを作成して、その後ドロップダウンリストから名前を選択するようにしてください。

4.2.3.7.2. エンドポイント

認証ポリシーが呼び出している mBaaS サービスによって定義されるエンドポイントの完全パス。

エンドポイントは、リクエストのボディにある [Auth API](#) で指定されている 'params' オブジェクトを受け取ります。認証が成功すると、200 が返されます。

承認が必要な場合は、このエンドポイントは **userId** キーとともに JSON 応答を返します。このキーの値は、承認チェックのためにシステムでユーザーを検索するために使用されます。

4.2.3.7.3. デフォルト環境

環境の値がリクエストにない場合に使用する環境。

4.2.3.8. 承認

ユーザー認証に加えて、承認方法も指定することができます。

4.2.3.8.1. ユーザー終了の確認

このチェックでは、ユーザーがプラットフォームを終了しているユーザーであることを確認します。プラットフォームでのユーザー管理については、ユーザーのセクションを参照してください。

4.2.3.8.2. ユーザーが承認済みか確認

ここでは、アプリへのアクセスを承認したい特定のユーザーを追加できます。このリストにユーザーがない場合は、アプリへのアクセスが許可されていません。

4.2.4. モバイルアプリの管理

RHMAP には、バイナリーをモバイルデバイスに配信するためにビルトインの App Store が含まれています。サードパーティーのモバイルアプリケーション管理ソリューションの統合もサポートされており、これは [Third Party MAM/MDM](#) セクションに記載されています。

4.2.4.1. App Store

App Store では、エンドユーザーが iOS、Android、および Windows 用のモバイルアプリをモバイルデバイスに直接インストールできます。

4.2.4.1.1. App Store の設定

各ドメインにはそれぞれ App Store があります。この App Store を使用するには、まず詳細を更新する必要があります。

- ※ 名前: App Store のユーザーに表示される名前
- ※ 説明: App Store のランディングページに表示される説明
- ※ App Store アイコン: App Store と一緒に表示されるアイコン
- ※ App Store アプリ: Store からインストール可能なアイテム
- ※ 認証ポリシー: Store にアクセス可能なユーザーの設定、およびインストール可能なアイテムの設定に使用します。

4.2.4.1.2. App Store の使用

App Store にアクセスするには、モバイルブラウザで App Store の URL に移動します。この URL は **管理者 > モバイルアプリの管理 > App Store** ページにあり、通常は **domain.feedhenry.com/store** となっています。

デバイスで App Store URL に移動すると、Mobile App Store のログイン画面が表示されます。この画面では App Store の詳細と、OAuth や LDAP などの利用可能な認証メカニズムの一覧が表示されます。これらのメカニズムは、本ガイドの他の部分を実行することで事前設定されます。認証メカニズムが設定されていない場合は、App Store にアプリが追加されていてもログインできません。

Mobile App Store



Mobile App Store

This is the mobile app store of your company and contains internal apps. This store is not related to the standard app store of your device manufacturer.

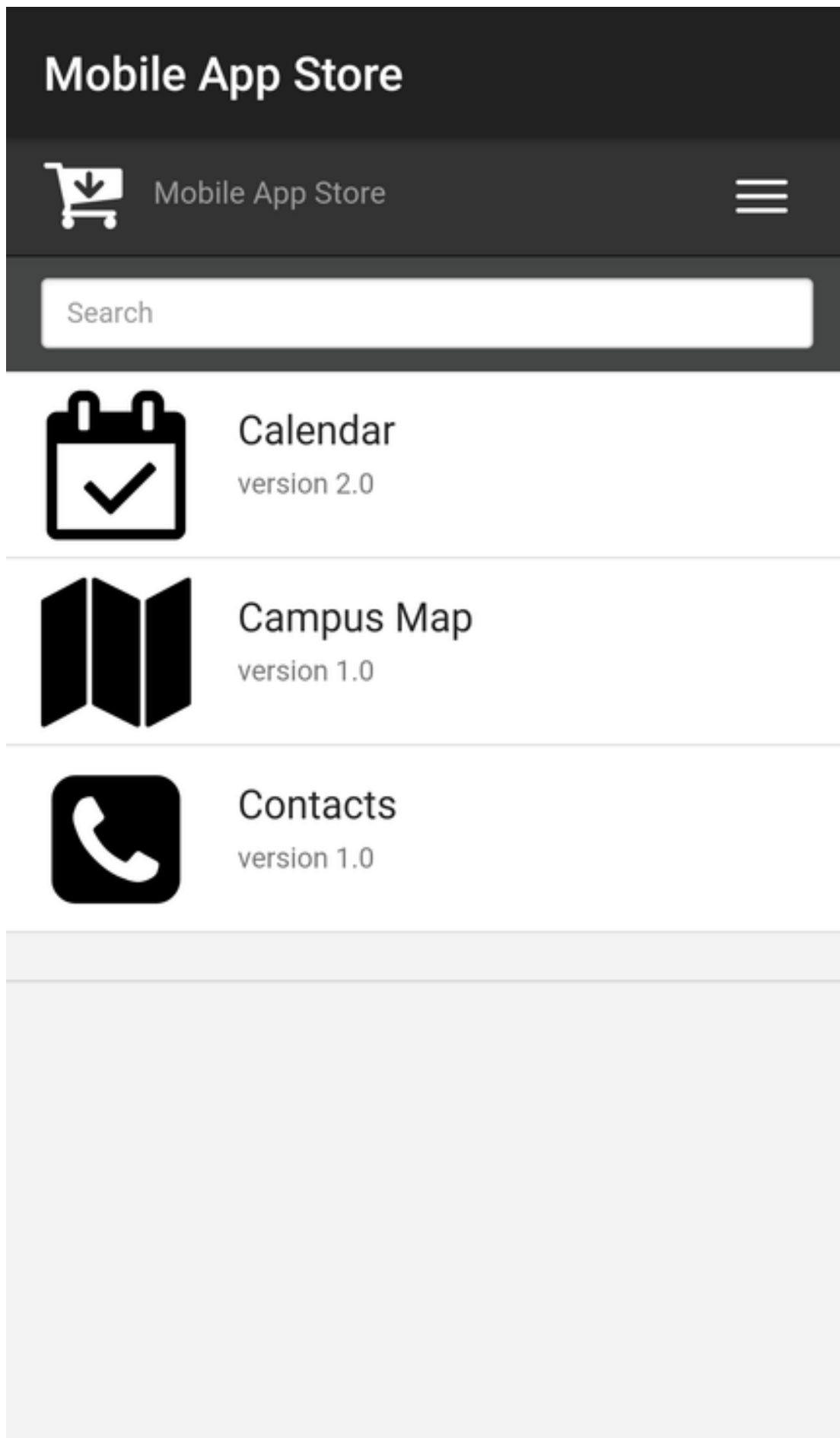
Log In

Sign in with LDAP

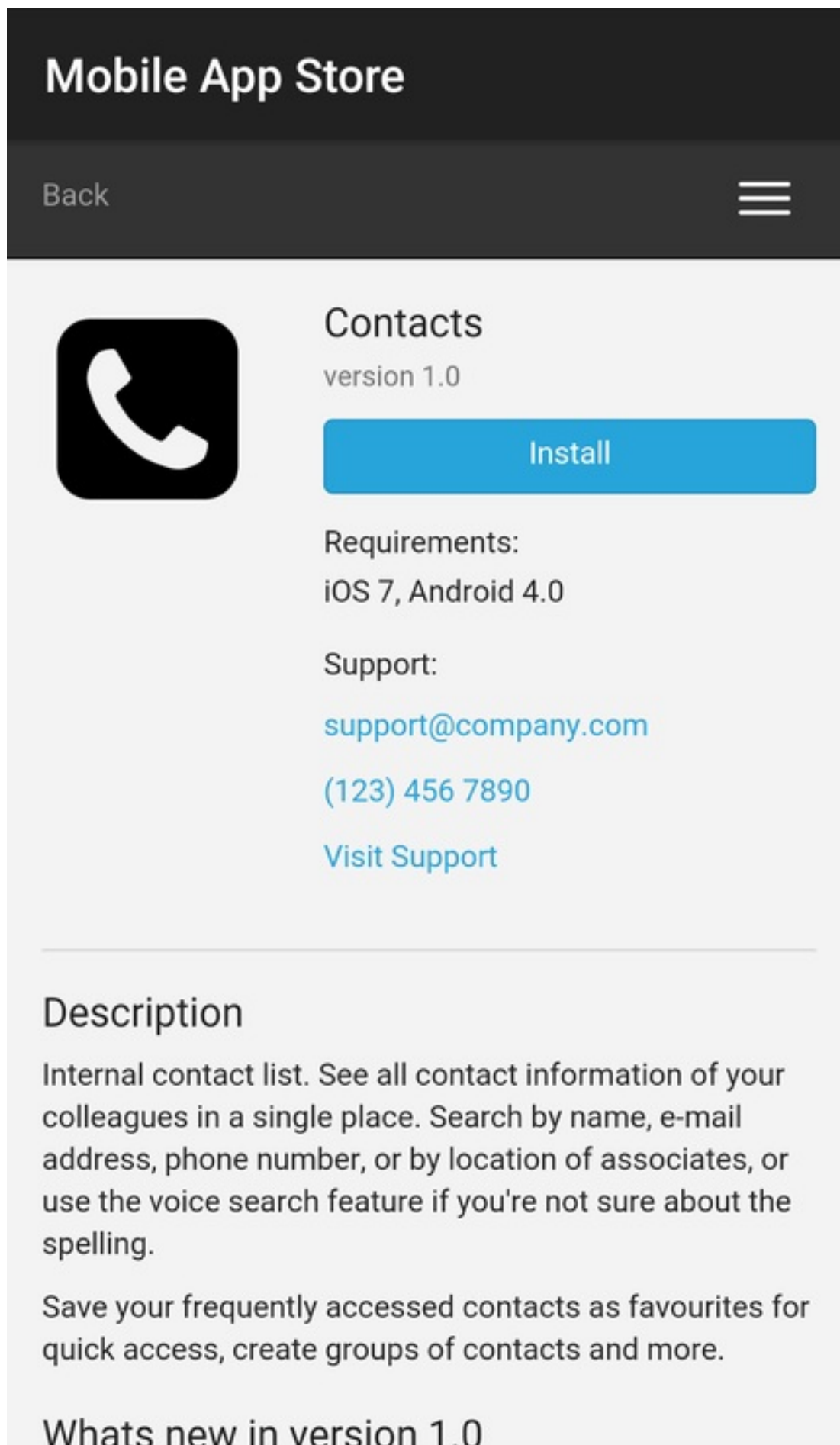
Sign in with FeedHenry

Sign in with OAuth

ログインすると、デバイスで使用可能なアプリが一覧表示されます。



アプリを選択すると、インストールボタンなどの詳細情報が表示されます。



iOS アプリケーションの場合は、以下のバイナリータイプがインストール可能です。

※ iPhone - iPhone、iPod Touch および iPad にインストール可能

- ※ iPad - iPad にインストール可能
- ※ iOS - 1つのパッケージで iPod Touch、iPhone および iPad にインストール可能なユニバーサルアプリケーション

利用可能なオプションは、アプリのビルド時に作成されたパッケージによって異なります。Android アプリの場合は通常、利用可能なインストールオプションは1つのみです。

4.2.4.2. App Store のアプリ

App Store のアプリ (以前の Store アイテム) は、App Store からの配信用に追加することができます。新規および既存の Android、iOS (ユニバーサル)、iPhone & iPad アプリ用にネイティブバイナリーをアップロードすることができます。また、App Store に表示されることになるアイコンを App Store アプリに割り当てたり、アプリに名前と説明を加えることもできます。

以下は、App Store のアプリで実行可能なアクションです。

- ※ 新規の App Store のアプリを配信用に追加。(通常はモバイルアプリ)
- ※ 新規または既存のアプリ向けにネイティブバイナリーをアップロード
- ※ 既存アプリのアプリバイナリー履歴を表示
- ※ 既存アプリのアプリバイナリー履歴の監査ログを表示
- ※ アプリケーションで使用する認証ポリシーを追加
- ※ アプリのアイコンを追加
- ※ 同一アプリのグループのユーザーのみにアプリを表示するように制限
- ※ アプリをアプリグループに追加

4.2.4.2.1. App Store のアプリの追加

新規アプリを作成するには、メニューで App Store のアプリにあるオプションをクリックします。名前フィールドがあるフォームが表示されます。このフィールドに入力して、「アプリの作成」ボタンをクリックします。

4.2.4.2.2. App Store のアプリ更新

アプリを更新するには、アプリの編集ボタンをクリックします。すると、アプリ更新ページの詳細タブが表示されます。

各種入力フィールドのあるフォームが表示されます。

- ※ 名前: アプリの一意の名前
- ※ アイコン: 「参照」ボタンをクリックします。使用するアイコンを選択して OK を押します。直ちにアイコンがアップロードされ、アプリの横に表示されます。
- ※ 認証トークン: 一意の文字列のトークンで、\$fh.auth API と合わせてアプリケーションを特定し、認証ポリシーで認証チェックを指定している場合は、これを実行します。
- ※ 説明: App Store でアプリの説明として表示されます。
- ※ 認証ポリシー: App Store のアプリで使用可能なポリシーが表示されます。このポリシーでは、アプリにアクセスを試みるユーザー認証に使用される認証方法が定義されます。

- ※ グループに制限しますか: このチェックボックスでは、アプリにおけるアプリグループの機能を制御します。チェックを入れると、ユーザーはアプリと同じアプリグループに属している必要があります。
- ※ アプリグループ: App Store のアプリで使用可能なアプリグループが表示されます。これらのグループは、同じアプリグループ (単数または複数) にいるユーザーに対するアイテムの視認性を制限します (「グループに制限しますか?」がチェックされている場合)。

必要な情報をこれらのフィールドに入力したら、「アイテム詳細の更新」ボタンをクリックします。



注記

これがアプリ作成後の初更新の場合、App Store でアプリが表示されるためにバイナリーの追加が必要だと Studio が判断して、バイナリータブが表示されます。

4.2.4.2.3. バイナリーのアップロード

バイナリーをアップロードする、または履歴を表示するには、既存のアプリが必要になります。

4.2.4.2.3.1. バイナリーのアップロード

既存の App Store アプリのいずれかで編集ボタンをクリックし、更新ページの「バイナリー」タブをクリックします。アプリのバイナリーをアップロードするには、プラットフォームまたは Xcode、もしくは Eclipse などビルドされた有効なバイナリーが必要になります。

アップロードするバイナリータイプを選択します。許可されるバイナリーの最大サイズは 64MB です。「参照」をクリックします。バイナリーファイルを選択すると、アップロードが開始されます。アップロードが完了すると、そのアップロードでダウンロード可能なオプションとなります。

4.2.4.2.3.2. iOS バイナリー

iOS バイナリーオプションの横には空白の設定フィールドがあります。このフィールドは、iOS アプリのバンドル識別子のためのものです。この識別子は、App Studio の設定の下の iPhone | iPad | iOS ユニバーサルにあります。Bundle Identifier のラベルが付けられています。

iOS ベースのアプリの場合、.ipa ファイルをアップロードする必要もあります。プラットフォームを使用してアプリを構築している場合は、iOS のビルド時にこのファイルをダウンロードできます。アプリを Xcode で構築している場合は、アーカイブオプションを使用すると .ipa ファイルが取得できます。

4.2.4.2.3.3. Windows Phone 8 バイナリー

ここでアップロードする Windows Phone 8 バイナリーは、企業証明書で署名する必要があります。バイナリーとともに AET ファイルを配信したい場合は、バイナリー用の設定ファイルとしてこれをアップロードすることができます。詳細は、[Deploying an App on Windows Phone 8](#) を参照してください。

4.2.4.2.3.4. バイナリーの履歴

既存の App Store アプリのいずれかで編集ボタンをクリックし、更新ページの「バイナリー」タブをクリックします。現行およびこれまでのバイナリーを表示するには、希望するバイナリーの「[+] 履歴」リンクをクリックします。これでバイナリー履歴一覧が表示されます。

4.2.4.3. サードパーティー MAM/MDM

個別のモバイルアプリケーション管理 (MAM) およびモバイルデバイス管理 (MDM) プロバイダーとの統合を有効または無効にできます。統合を有効にしてもすべてのバイナリーが自動的に公開されるわけではありません。公開を有効にする方法については、[Publishing App Binaries to Third-party MAM and MDM providers](#) を参照してください。

4.2.5. 監査ログ

監査ログでは、アプリバイナリーのダウンロード数が分かります。監査ログを表示するには、既存のアプリが必要になります。

既存の App Store アプリのいずれかで編集ボタンをクリックし、更新ページの「監査ログ」タブをクリックします。これでそのアプリの現行の監査ログエントリーが表示されます。ページ左側で「モバイルアプリの管理」から「監査ログ」をクリックすると、全アプリの完全な監査ログエントリー一覧が処理されて表示されます。

4.2.5.1. 検索とフィルターの機能

検索ボックスを使用すると監査ログを検索できます。タブ上部のオプションを使用すると、監査ログを以下でフィルタリングできます。

- ※ アプリ名
- ※ ユーザーのメールアドレス
- ※ アプリタイプ (iOS、iPhone、iPad または Android)
- ※ 表示するレコード数 (10、100 または 1000)

フィルターオプションを選択したら、「フィルター」ボタンをクリックして一致する監査ログを表示します。「リセット」ボタンをクリックすると、デフォルトの監査ログが表示されます。

4.2.6. グループ

App Store のグループ (以前は Store アイテムグループ) を作成すると、アプリを特定のユーザーグループに制限できます。アプリを特定のグループの集合に制限するようにマークを付けると、そのグループのユーザーのみが App store にあるこれらのアイテムを見ることができます。

App Studio の アプリグループ管理セクションでは、管理者は以下のことができます。

- ※ 新規アプリグループの作成
- ※ ユーザーのグループへの割り当て
- ※ App Store のアプリのグループへの割り当て

4.2.6.1. 概要

ユーザーと App Store のアプリを作成します。App Store のアプリの作成時に、これをグループに制限するようマークを付けることができます。ただし、この段階ではグループが作成されていない場合もあるので、選択する必要はありません。

ユーザーと App Store のアプリを作成したら、アプリグループを作成することができます。作成するグループは組織内の部門に対応させたりします。販売部門用のアプリを含む販売グループや、エンジニアリング部門用のアプリを含むエンジニアリンググループなどです。グループを作成したら、App Store のアプリとユーザーをグループに割り当てられます。

例えば、販売ユーザーと販売 App Store アプリを販売グループに割り当てると、販売ユーザーの 1 人が App Store にログインした場合、販売 App Store アプリを見ることができます。ユーザーと App Store アプリは複数のグループに割り当てることができるので、販売とエンジニアリングのマネージャーが両方のグループに属し、両部門の App Store アプリにアクセスするということが可能になります。もちろん App Store アプリをグループに制限せず、App Store にログインした全ユーザーに表示されるようにすることもできます。

4.2.6.2. 新規グループの追加

新規のアプリグループを作成するには、メニューのモバイルアプリの管理セクションでグループオプションをクリックし、「グループの作成」ボタンをクリックします。2つの入力フィールドがあるフォームが表示されます。

- ✎ 名前: アプリグループの一意の名前
- ✎ 説明: グループの目的が分かるように説明を入力します。

必要な情報をこれらのフィールドに入力したら、「グループの作成」ボタンをクリックします。

4.2.6.3. グループの編集

グループは編集して説明のテキストを変更したり、ユーザーもしくはアプリをグループに割り当てることができます。アプリグループを編集するには、メニューのモバイルアプリの管理セクションでグループオプションをクリックし、編集するグループの「編集」ボタンをクリックします。

- ✎ App Store Apps: グループに現在割り当てられているアイテム
- ✎ ユーザー: グループに現在割り当てられているユーザー

4.2.7. デバイス

アクティブなデバイスは表示やフレンドリー名の割り当て、ブロックやデータ削除をマークすることができます。デバイス上のアクティブなアプリとユーザーも分かります。

以下が実行可能なアクションになります。

- ✎ アクティブなデバイスの表示
- ✎ デバイス ID へのフレンドリー名の割り当て
- ✎ デバイス上で使用中のアプリを表示
- ✎ デバイス上でアクティブなユーザーを表示
- ✎ デバイスをブロック - デバイスからアプリやユーザーが認証できないようにします。
- ✎ デバイスデータの完全削除 - アプリの次の認証試行時に、デバイス上のアプリに kill pill シグナルを送信します。

4.2.7.1. デバイスの表示

デバイス一覧表示では、\$fh.auth を使用してアプリにアクセスした全デバイスが表示されます。一覧では、デバイスラベル、デバイス ID、デバイスのステータス、最新のアクセス日時が表示されます。いずれかの制御ボタンをクリックすると、デバイスの更新ビューが表示されます。

デバイス ID はデバイスの一意の識別子で、変更はできません。これはデバイスから送信され、同一デバイス (ハイブリッドまたはネイティブアプリ) 上の異なるアプリで同じものになります。唯一の例外は、ウェブブラウザで読み込まれたウェブアプリです。この場合、デバイス ID は生成された値で、クッキーに保存されます。クッキーが削除されると、新たな値が生成されます。

4.2.7.2. デバイスの更新

このビューでは、デバイスにフレンドリー名を割り当てたり、認証を無効にしたり、データの完全削除をマークしたりできます。また、プラットフォーム上のどのユーザーがデバイスを使用しているか、およびそのデバイスからアクセスしているアプリを確認できます。

デバイスの認証を無効にすると、\$fh.auth を使用して認証しているデバイス上の全アプリが失敗するようになり、デバイスが無効になっていることを示す特別な値が返されます。これは、デバイスを紛失したり盗難にあった場合などに便利です。

「デバイスデータを完全削除しますか?」のオプションが選択されると、以下のことが実行されます。

- ※ デバイスの認証が無効になります。
- ※ デバイス上のアプリが \$fh.auth を使用して認証すると、認証プロセスが失敗し、既存の全アプリデータを削除すべきという特別な値を API が返します (データ削除機能の実装は、開発者の判断になります)。

4.2.7.3. デバイスユーザーの表示

\$fh.auth がデバイス上のアプリから呼び出されるたびに、認証プロセスが成功してユーザーがプラットフォームユーザーであれば、アクセスログエントリが作成されます。このため、デバイス表示エリアでは、どのユーザーがどのデバイスからアクセスしたかを確認することができます。

ユーザーセクションでは、そのデバイスを使用してアクセスしたユーザーの ID 一覧が表示されます。ユーザー ID にカーソルを移動させるとユーザーについての詳細が示され、クリックするとそのユーザーの管理画面に移動します。

4.2.7.4. デバイス上のアプリの表示

ユーザーアクセスの場合と同様に、\$fh.auth を呼び出すたびにアプリ情報もキャプチャーされます。アプリ一覧は、デバイス表示エリアの "App" セクションに表示されます。各アプリエントリーにカーソルを移動させると、アプリの概要が表示され、クリックするとそのアプリの管理画面に移動します。

4.3. アナリティクス

概要

App Studio では、集計レポート (ドメイン内の全プロジェクトのレポート) と個別レポートの両方が利用可能です。これらのレポートには、アナリティクスメニューからアクセスできます。また、集計レポートは、個別のプロジェクトごとに分割することができます。

要件

ユーザーは、以下のパーミッションを持つ 1 つ以上のチームメンバーである必要があります。

- ✦ ドメイン — プロジェクト (表示)
- ✦ ドメイン — アナリティクス (表示)

関連リソース

- ✦ [Cloud API - Statistics API](#)

4.3.1. 集計レポート

集計レポートは、選択した期間のドメイン内の全レポートの高度な累積ビューを提供します。例えば、アクティブなデバイスの中央値やアプリインストールの合計数などです。

4.3.1.1. 月間アクティブデバイス

これは、選択した 31 日間の期間中にクラウドリクエストを行ったデバイス数です。[一意のデバイス識別子](#)で特定されるデバイスは、この期間に少なくとも 1 回のクラウドリクエストを`$fh.act`または`$fh.cloud`で行うと、アクティブとみなされます。

4.3.1.1.1. デバイス識別子

デバイスは、クライアントの一意識別子 (CUID) フィールドで特定されます。これは、RHMAP クライアント SDK により、各クラウドリクエストに自動的に追加されます。

CUID は各プラットフォームで異なる API を使用して生成されます。

- ✦ iOS では、[Advertising Identifier](#) を使用して CUID の値が生成されます。
- ✦ Android では、[Android ID](#) を使用して CUID の値が生成されます。
- ✦ JavaScript SDK を使用するブラウザーでは、CUID はブラウザーに保存される (利用可能なストレージメカニズムで保存) ランダムな文字列になります。アプリデータが削除されると、新たな CUID が生成されます。



注記

各プラットフォームの制限のために、CUID の値が特定のデバイス上で一定のものに留まることは保証されません。デバイスのオペレーティングシステムによっては、デバイスの再起動やアプリの再インストール、デバイスからアプリデータを削除することでこの値が変化する可能性があります。

iOS の場合、同一デバイス上でもアプリによって CUID の値は異なります。つまり、各アプリでは同一デバイスを新規デバイスとしてカウントすることになります。iOS プラットフォームでは、この制限があります。

4.3.2. プロジェクト別レポート

プロジェクトでは、デバイスのインストール、アプリの起動、クラウドのリクエスト、およびアクティブなデバイスという 4 つのレポートタイプがあります。ドメインの設定によっては、クラウドリクエストとアクティブデバイスは利用可能でない場合もあります。

4.3.2.1. デバイスインストール

デバイスインストールでは、特定の期間にアプリがインストールされた数が表示されます。デバイスインストールは、アプリが新規デバイスで最初に起動し、クラウドにコールバックをすると追跡されます。デバイスインストールは App Stores が以下の方法で追跡するユーザーインストールとは異なることに注意してください。

- ※ App Stores では、ユーザーがアプリをインストールしたデバイス数にかかわらず、インストールをユーザーアカウントのダウンロード実行として数えます。これは、個別の App Store から利用可能なレポートタイプとは異なる可能性があります。
- ※ App stores は通常 PST タイムゾーンで日次結果をレポートします (Google Play および iTunes Connect の場合) が、デバイスインストールでは UTC でレポートされます。
- ※ アプリインストールが追跡されるには、アプリが有効なインターネット接続で少なくとも 1 回は起動される必要があります。ここでインストールがカウントされるのは、ダウンロード時ではなく、アプリの初回起動時です。

4.3.2.2. アプリ起動

アプリの起動は、インターネット接続でこれまでに見られたデバイス上でアプリが起動されると追跡されます。アプリ起動は、コールドスタートで起動したアプリのみが追跡されることに注意してください。つまり、バックグラウンドで実行されているアプリがフォアグラウンドに持ち込まれても「アプリ起動」として追跡されません。

4.3.2.3. クラウドリクエスト

クラウドリクエストのレポートでは、クライアントアプリが (**\$fh.cloud** を使って) クラウドアプリにリクエストを要求した回数が表示されます。

4.3.2.4. アクティブデバイス

アクティブデバイスのレポートは、選択した期間であるアプリに対してアクティブだったデバイス数の中央値を示します。[一意のデバイス識別子](#)で特定されるデバイスは、この期間に少なくとも 1 回のクラウドリクエストを (**\$fh.act** または **\$fh.cloud** で) 24 時間の期間 (GMT タイムゾーンの午前 0 時から) の間に行うと、アクティブとみなされます。

4.3.3. レポート図表 & グラフ

各タイプのレポートは、希望するデータがどの図表で一番うまく表示されるかによって、多くの方法で見ることができます。一般的には、個別の図表で示されている情報は、カーソルをその上に移動すると詳しい説明が表示されます。ある図表で複数のアイテムが比較されている場合は、凡例でそのアイテムをクリックしてオン、オフを切り替えることができます (ほとんどの図表で対応)。図表のデータは、個別の図表に関連付けられているエクスポートボタンをクリックすることで、CSV、PNG または PDF 形式でエクスポートできます。

図表の日付の範囲は変更可能で、カレンダーで範囲を選択するか、例えば「過去 7 日間」「過去 30 日間」「前月」から選択することもできます。

図表のデータは毎日更新され、最新のデータは前日のものになります。

4.3.3.1. 集計レポート

レポートタブの集計レポートは、全レポートの高度なビューを日付、デバイス、場所ごとに提供します。各アナリティクスタイプで「詳細を表示」をクリックすると、簡単に特定のレポートをドリルダウンすることができます。

4.3.3.2. アプリ別レポート

レポートタブのアプリ別レポートでは、左側のリストからアプリを選択して、個別アプリについてのレポートを表示することができます。多数のアプリがある場合は、検索することも可能です。アプリを選択したら、このアプリについての概要レポートが表示されます。特定のメトリクスについて知りたい場合は、「詳細を表示」ボタンをクリックします。

4.3.3.3. 個別アプリの分析

個別アプリの画面では、特定タイプのアナリティクス (起動、インストールなど) についての概要レポートが表示されます。レポートのタイトルをクリックするか、ナビゲーションで表示を切り替えると個別の図表が表示されます。

4.3.3.4. 日付別の分析

このタイプの図表では、時間経過による値の推移が分かります。Y 軸にはインストールなどの値が表示され、X 軸では日付が示されます。各プラットフォーム (Android、iPhone など) の値は、個別の線で示されます。

4.3.3.5. プラットフォーム別の分析

プラットフォーム別の集計値が円グラフで確認できます。集計値は、選択された期間のものです。

4.3.3.6. 場所別の分析

ユーザーの場所をベースにした特定期間の集計値が地理的グラフで表示されます。国の上にカーソルを移動すると、国ごとの値や、最小から最大値の基準におけるその値の位置が表示されます。

4.4. チームおよびコラボレーション

4.4.1. 概要

チームおよびコラボレーションのセクションでは、Red Hat Mobile Application Platform ホスト型 (RHMAP) 内の特定の機能およびエンティティーへのアクセスを制限することができます。

このセクションでは、以下を実行できます。

- ✦ 新規チームの作成
- ✦ ユーザーを 1 つ以上のチームに追加する。ユーザーをあるチームに追加すると、そのチームのメンバーになります。チームに割り当てられているパーミッションはすべて、このユーザーに適用されるようになります。
- ✦ プラットフォームの特定機能へのアクセスを制限するために、チームにパーミッションを割り当てる。

チームは以下の 2 つのカテゴリに分けられます。

1. **デフォルトチーム** は、プラットフォームが作成したチームです。デフォルトチームのパーミッションと詳細 (名前や説明など) は編集できません。デフォルトチームにはメンバーの追加および削除のみが実行可能です。

2. **User Created Teams** (ユーザー作成のチーム) は、ユーザーが作成したチームです。User Created Team の詳細とパーミッションは編集可能です。

4.4.1.1. チームリスト

この画面では、現在利用可能なチームが一覧表示されます。以下が実行可能です。

- ※ チーム一覧をグリッドまたは表形式で表示する。
- ※ チーム名での検索
- ※ **顧客** および**ドメイン** でチーム一覧にフィルターをかける。
- ※ **チームの作成** ボタンを選択して新規チームを作成する。

4.4.2. チームの作成

名前 と **説明** フィールドに入力し、**チームの作成** ボタンをクリックするとチームが作成されます。

4.4.3. ダッシュボード

ダッシュボード画面では、チームの詳細が表示されます。この画面では、以下が実行できます。

- ※ チームに現在割り当てられているユーザーの一覧表示。これらのユーザーは、チームの **メンバー** と呼ばれます。
- ※ **メンバーの編集** リンクまたは画面左側にある**メンバー** タブをクリックして、**メンバー** 一覧を編集する。
- ※ チームに現在割り当てられている **パーミッション** を表示する。
- ※ **パーミッションの編集** リンクまたは画面左側にある**パーミッション** タブをクリックして、チームに割り当てられている **パーミッション** を編集する。
- ※ チームの名前と説明を編集する。
- ※ **このチームを削除** ボタンを選択して、現在選択しているチームを削除する。



注記

デフォルトチームの **名前** と **説明** は編集できません。



注記

デフォルトチームは削除できません。

4.4.4. メンバー

メンバーセクションでは、チームのメンバー管理ができます。この画面では、以下が実行できます。

- ※ チームの現行メンバーの一覧表示

- ✧ テキストエリアをクリックし、ユーザーを選択してチームにユーザーを選択する。選択されたユーザーは、自動的にチームのメンバーとして追加されます。
- ✧ チームのメンバーの横にある「削除」ボタンを選択してメンバーを削除する。
- ✧ **編集** ボタンを選択して、チーム名と説明を編集する。



注記

ユーザーは複数のチームの **メンバー** となることが可能です。

4.4.5. パーミッション

パーミッションセクションでは、現在選択しているチームに関連するパーミッションの設定ができます。チームの全 **メンバー** がこのチームに関連付けられているパーミッションに制限されます。

4.4.5.1. 主要な用語

パーミッションに関連する主な用語は以下の通りです。

- ✧ **ビジネスオブジェクト**: プラットフォーム内のエンティティを説明するラベル (例: プロジェクト)。
- ✧ **エンティティ**: ビジネスオブジェクトのインスタンス (例: **My First Project** という名前のプロジェクトは、プロジェクトビジネスオブジェクトのインスタンス。) **パーミッション**: ビジネスオブジェクトに適用されるアクセスレベル (例: 読み取り、書き込み)。
- ✧ **チーム**: 選択したビジネスオブジェクトに適用されるパーミッションの集合。
- ✧ **階層**: 相互のビジネスオブジェクトにおける位置を示し、UI の機能エリアも示します。

4.4.5.2. ビジネスオブジェクト

パーミッションはすべて、ビジネスオブジェクトに対して設定されます。

ビジネスオブジェクトはプラットフォームの機能エリアに関連し、その構造を示す階層に体系化されます。

例えば、**リセラー** に下にある **管理者** 要素にパーミッションを設定すると、この **リセラー** の管理者機能に対するパーミッションを設定することになります。

4.4.5.3. アクセスレベル

ビジネスオブジェクトには、以下の 3 つのアクセスレベルを適用できます。

- ✧ **No Access**: **チーム** はこのビジネスオブジェクトにアクセスできません。
- ✧ **表示**: **チーム** はビジネスオブジェクトを表示することはできますが、ビジネスオブジェクトに変更を加えることはできません。
- ✧ **表示 & 編集**: **チーム** はビジネスオブジェクトを表示でき、かつ変更を加えることもできます。

4.4.5.4. 継承

ビジネスオブジェクトは階層に体系化されます。ビジネスオブジェクトは、プラットフォームの大きな機能エリアを参照することができます (例: **ドメイン** 全体)。サブビジネスオブジェクトは、プラットフォームのより特定の機能エリアを参照します (例: **ドメイン** 内の **プロジェクト**)。

この構造を使用すると、サブビジネスオブジェクトの **パーミッション** をその親から継承できます。これは、**パーミッションを継承** の選択肢で決定します。**パーミッションを継承** を **オン** にすると、サブビジネスオブジェクトはその親のパーミッションを継承します。**パーミッションを継承** を **オフ** にすると、特定の **アクセスレベル** をサブビジネスオブジェクトに設定する必要があります。



注記

親レベルの **パーミッションを継承** が **オン** に設定されていれば、ビジネスオブジェクトは 2 つ上またはそれ以上のレベルの **パーミッション** を継承できます。

パーミッション の優先順位は、ビジネスオブジェクトに設定された **アクセスレベル** で決定されます。

ビジネスオブジェクトに **アクセスレベル** を設定する順番は、以下のようになります。

1. 読み取り & 書き込み
2. 表示
3. No Access
4. パーミッションを継承

4.4.5.5. 複数チーム

ユーザーは、複数の **チーム** の **メンバー** となることができます。この場合、ビジネスオブジェクトに適用される **アクセスレベル** は、上記の優先順位となります。

以下の例を見てみましょう。

- ※ チーム A でビジネスオブジェクト A に **パーミッション継承** の **アクセスレベル** を設定します。
- ※ チーム B ではビジネスオブジェクト A に **表示** の **アクセスレベル** を設定します。
- ※ ユーザー A はチーム A とチーム B のメンバーです。

このシナリオでは、ビジネスオブジェクト A には **表示** の **アクセスレベル** があることとなります。

4.4.5.6. フィルタリング可能なビジネスオブジェクト

ビジネスオブジェクトには、複数のインスタンスがあるものもあります。例えば、**プロジェクト** ビジネスオブジェクトは、ドメイン上で利用可能なすべての **プロジェクト** に関係します。ドメインでは複数のプロジェクトが存在する場合があるので、特定のプロジェクトにパーミッションを適用する必要があります。

フィルタリング可能なビジネスオブジェクトは、以下の 2 つの方法でフィルターをかけられます。

1. **すべてにアクセス** を **オン** に設定すると、そのチームはビジネスオブジェクトの全インスタンスにアクセスできるようになります。例えば、**プロジェクト** ビジネスオブジェクトで **すべてにアクセス** を **オン** に設定すると、そのチームはドメイン内の全プロジェクトにアクセスできるようになります。
2. **すべてにアクセス** を **オフ** に設定すると、チームがアクセス可能なビジネスオブジェクトのインスタンスを選択できるようになります。

ユーザーが作成したビジネスオブジェクトのインスタンスはすべて、そのユーザーには表示されます。ただし、そのビジネスオブジェクトに割り当てられているパーミッションが適用されます。

以下の例を見てみましょう。

- ※ ユーザー A は チーム A のメンバー です。
- ※ **プロジェクト** ビジネスオブジェクトの **アクセスレベル** を **読み取り & 書き込み** に設定します。
- ※ ユーザー A が **プロジェクト A** という名前の新規 **プロジェクト** を作成します。
- ※ チーム A について、**プロジェクト** ビジネスオブジェクトの **すべてにアクセス** を **オフ** に設定します。
- ※ **プロジェクト** ビジネスオブジェクトでインスタンスをなにも選択しません。

このシナリオでは、Studio のプロジェクトセクションでユーザー A に表示されるのは、**プロジェクト A** のみになります。